

Byoung-Tak Zhang
Mehmet A. Orgun (Eds.)

LNAI 6230

PRICAI 2010: Trends in Artificial Intelligence

11th Pacific Rim International Conference on Artificial Intelligence
Daegu, Korea, August/September 2010
Proceedings

 Springer

REPORT DOCUMENTATION PAGE

Form Approved

OMB No. 0704-0188

The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE (DD-MM-YYYY) 08-09-2010		2. REPORT TYPE Conference Proceedings		3. DATES COVERED (From - To) 30-Aug-10 – 02-Sep-10	
4. TITLE AND SUBTITLE PRICAI 2010: The 11th Pacific Rim International Conference on Artificial Intelligence				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER FA23861011038	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S) Byoung-Tak Zhang and Mehmet A. Orgun (Eds.)				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) School of Computer Science and Engineering, Kyungpook National University Sankyunk-Dong 1370, Buk-Gu Daegu 702-701 Korea				8. PERFORMING ORGANIZATION REPORT NUMBER N/A	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) AOARD UNIT 45002 APO AP 96338-5002				10. SPONSOR/MONITOR'S ACRONYM(S) AOARD	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S) CSP-101038	
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; U.S. government purpose rights.					
13. SUPPLEMENTARY NOTES Springer ©2010, Springer-Verlag; Berlin The U.S. Government has a non-exclusive license rights to use, modify, reproduce, release, perform, display, or disclose these materials, and to authorize others to do so for US Government purposes only. All other rights reserved by the copyright holder.					
14. ABSTRACT PRICAI is a biannual conference on Pacific Rim's artificial intelligence conference. There were 69 papers accepted, out of which 48 were orally presented and 21 were poster-presented. This volume contains these 69 papers plus summaries of 1 key note speech and 3 invited talks. The topics covered include AI foundations, Applications of AI, Agents, Bioinformatics, Cognitive modeling and human interaction, Computer-aided education, Constraint satisfaction, Creativity support, Decision theory, Evolutionary computation, Game playing and interactive entertainment, Heuristics, Information integration and extraction, Information retrieval and extraction, Knowledge acquisition and ontology, Knowledge representation, Machine learning and data mining, Model-based systems, Multimedia and AI, Natural language processing, Planning and scheduling, Reasoning, Robotics, Text/Web data mining, Social intelligence, Speech processing, Uncertainty, and Vision and perception.					
15. SUBJECT TERMS Artificial Intelligence, Machine Learning, Data Mining, Natural Language Processing, Agent Based Modeling					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Hiroshi Motoda, Ph. D.
U	U	U	UU	715	19b. TELEPHONE NUMBER (Include area code) +81-3-5410-4409

Lecture Notes in Artificial Intelligence

6230

Edited by R. Goebel, J. Siekmann, and W. Wahlster

Subseries of Lecture Notes in Computer Science

Byoung-Tak Zhang Mehmet A. Orgun (Eds.)

PRICAI 2010: Trends in Artificial Intelligence

11th Pacific Rim International Conference
on Artificial Intelligence
Daegu, Korea, August 30 – September 2, 2010
Proceedings

20101130209



Springer

Series Editors

Randy Goebel, University of Alberta, Edmonton, Canada
Jörg Siekmann, University of Saarland, Saarbrücken, Germany
Wolfgang Wahlster, DFKI and University of Saarland, Saarbrücken, Germany

Volume Editors

Byoung-Tak Zhang
School of Computer Science and Engineering
Seoul National University
Seoul, Korea
E-mail: btzhang@bi.snu.ac.kr

Mehmet A. Orgun
Department of Computing
Macquarie University
Sydney, NSW, Australia
E-mail: mehmet.orgun@mq.edu.au

Library of Congress Control Number: 2010932614

CR Subject Classification (1998): I.2, H.3, H.4, F.1, H.2.8, J.3

LNCS Sublibrary: SL 7 – Artificial Intelligence

ISSN 0302-9743
ISBN-10 3-642-15245-7 Springer Berlin Heidelberg New York
ISBN-13 978-3-642-15245-0 Springer Berlin Heidelberg New York

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer. Violations are liable to prosecution under the German Copyright Law.

springer.com

© Springer-Verlag Berlin Heidelberg 2010
Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper 06/3180

Preface

This volume contains the papers presented at The 11th Pacific Rim International Conference on Artificial Intelligence (PRICAI 2010) held during August 30–September 2, 2010 in Daegu, one of the most dynamic urban cities in Korea with a rich traditional cultural heritage.

PRICAI is a biennial conference inaugurated in Tokyo in 1990 to promote collaborative exploitation of artificial intelligence (AI) in the Pacific Rim nations. Over the past 20 years, the conference has grown, both in participation and scope, to be a premier international AI event for all major Pacific Rim nations as well as the countries from all around the world, highlighting the most significant contributions to the field of AI. This year, PRICAI 2010 also featured several special sessions on the emerging multi-disciplinary research areas ranging from *Evolving Autonomous Systems* to *Human-Augmented Cognition*.

There was an overwhelming interest to the call for papers for the conference. As a result, PRICAI 2010 attracted 191 full-paper submissions to the regular session and the special sessions of the conference from researchers from many regions of the world. Each submitted paper was carefully considered by a combination of several independent reviewers, Program Committee members, Associate Chairs, Program Vice Chairs and Program Chairs, and finalized in a highly selective process that balanced many aspects of the paper, including the significance, originality, technical quality and clarity of the contributions, and its relevance to the conference topics. As a result, this volume reproduces 48 papers that were accepted as regular papers (including the special session papers) and 21 papers that were accepted as short papers. This gives a regular paper acceptance rate of 25.13%, and a short paper acceptance rate of 10.99%, with an overall paper acceptance rate of 36.12%.

The regular papers were presented over three days in the topical program sessions and special sessions during August 31–September 2. The short papers were presented in an interactive poster session, as well as in a plenary session, contributing to a stimulating conference for all the participants. The PRICAI 2010 program also featured *The 11th International Workshop on Knowledge Management and Acquisition for Smart Systems and Services (PKAW 2010)* chaired by Paul Compton (University of New South Wales, Australia) and Hiroshi Motoda (Osaka University, Japan). The PKAW series has been an integral part of the PRICAI program over the past 11 years and this year was no exception.

We were also honored to have keynote presentations by four distinguished researchers in the field of AI whose contributions have crossed discipline boundaries: Heinrich Bülthoff from Max Planck Institute for Biological Cybernetics, Germany, talked on *Towards Artificial Systems: What Can We Learn from Human Perception?*; Mitsuru Ishizuka from University of Tokyo, Japan, on *Exploiting Macro and Micro Relations Toward Web Intelligence*; Mike

Schuster from Google, USA, on *Speech Recognition for Mobile Devices at Google*; and Toby Walsh from NICTA, Australia, on *Symmetry Within and Between Solutions*. We were grateful to them for sharing their insights on their latest research with us.

The PRICAI 2010 program was the culmination of efforts expanded so willingly by numerous people from all over the world over the past year. We would like to thank all the Program Vice Chairs and the Associate Chairs for their extremely hard work in the review process and the Program Committee members and the reviewers for a timely return of their comprehensive reviews of the submitted papers. Without their help and expert opinions, it would have been impossible to make decisions on each submitted paper and produce such a high-quality program. We would like to acknowledge the contributions of all the authors of the 191 submissions who made the program possible in the first place.

We would like to thank the Conference General Chairs, Jin-Hyung Kim (KAIST, Korea) and Abdul Sattar (Griffith University, Australia) for their continued support and guidance, and the Organizing Chairs Seong-Bae Park (Kyungpook National University, Korea) and Cheol-Young Ock (University of Ulsan, Korea) for making sure that the conference ran smoothly. Thanks are also due to:

- *Special Sessions Chairs*: Bob McKay, Minho Lee and Michael Strube
- *Tutorials Chairs*: Zhi-Hua Zhou and Kee-Enng Kim
- *Workshops Chairs*: Aditya Ghose and Shusaku Tsumoto
- *Posters Chairs*: Sanjay Chawla and Kyu-Baek Hwang
- *Publications Chair*: Byeong-Ho Kang
- *Treasury Chair*: Bo-Yeong Kang
- *Publicity Chairs*: Jung-Jin Yang, Takayuki Ito, Zhi Jin and Pau Scerri

Microsoft's CMT conference management system was used in all stages of the paper submission and review process and also in the collection of the final camera-ready papers; it made our life much easier.

We also greatly appreciated the financial support from Air Force Office of the Scientific Research/Asian Office of Aerospace Research and Development (AFOSR/AOARD), Office of Naval Research Global (ONRG), National Research Foundation of Korea, ETRI, LG CNS, KT, Soongsil University, Soft on Net, Saltlux, CRH for Human, Cognition and Environment, Daegu Convention & Visitors Bureau, and Korea Tourism Organization.

Special thanks go to Min Su Lee (Seoul National University, Korea) for supporting the committees so effectively; her dedication and resourcefulness made all the difference at several critical junctions of the whole process.

It has been a great pleasure for us to serve as the Program Chairs of PRICAI 2010 and to present a high-quality scientific program for the benefit of the participants of the conference as well as the readers of this proceedings volume.

September 2010

Byoung-Tak Zhang
Mehmet A. Orgun

Organization

PRICAI 2010 was hosted and organized by The Korean Institute of Information Scientists and Engineers, The Korean Society for Cognitive Science, and Kyungpook National University. The conference was held at Novotel Daegu, in Daegu, Korea, during August 30–September 2, 2010.

Steering Committee

Standing Members

Abdul Sattar	Griffith University, Australia (Chair)
Zhi-Hua Zhou	Nanjing University, China (Secretary)
Hideyuki Nakashima	Future University - Hakodate, Japan
Mitsuru Ishizuka	University of Tokyo, Japan
Geoff Webb	Monash University, Australia
Chengqi Zhang	University of Technology, Sydney, Australia
Tru Hoang Cao	Vietnam National University, Vietnam
Tu-Bao Ho	JAIST, Japan

Honorary Members

Hiroshi Motoda	Osaka University, Japan
Randy Goebel	University of Alberta, Canada
Wai K. Yeap	Auckland University of Technology, New Zealand

Organizing Committee

General Chairs

Jin-Hyung Kim	KAIST, Korea
Abdul Sattar	Griffith University, Australia

Program Chairs

Byoung-Tak Zhang	Seoul National University, Korea
Mehmet A. Orgun	Macquarie University, Australia

Organizing Chairs

Seong-Bae Park	Kyungpook National University, Korea
Cheol-Young Ock	University of Ulsan, Korea

Special Sessions Chairs

Bob McKay	Seoul National University, Korea
Minho Lee	Kyungpook National University, Korea
Michael Strube	EML Research, Germany

Tutorials Chairs

Zhi-Hua Zhou	Nanjing University, China
Kee-Eung Kim	KAIST, Korea

Workshops Chairs

Aditya Ghose	University of Wollongong, Australia
Shusaku Tsumoto	Shimane University, Japan

Posters Chairs

Sanjay Chawla	University of Sydney, Australia
Kyu-Baek Hwang	Soongsil University, Korea

Publications Chair

Byeong-Ho Kang	University of Tasmania, Australia
----------------	-----------------------------------

Treasury Chair

Bo-Yeong Kang	Kyungpook National University, Korea
---------------	--------------------------------------

Publicity Chairs

Jung-Jin Yang	Catholic University, Korea
Takayuki Ito	NIT/MIT, Japan/USA
Zhi Jin	Peking University, China
Pau Scerri	Carnegie Mellon University, USA

Program Committee

Chairs

Byoung-Tak Zhang	Seoul National University, Korea
Mehmet A. Orgun	Macquarie University, Australia

Vice Chairs

Hung H. Bui	Doheon Lee
Vladimir Estivill-Castro	Thomas Meyer
Eibe Frank	Wee Keong Ng
Tu-Bao Ho	Maurice Pagnucco
Achim Hoffmann	Mikhail Prokopenko
Wynne Hsu	Leon Sterling
Ryszard Kowalczyk	Markus Stumptner
James Kwok	Kewen Wang

Qiang Yang
Chengqi Zhang

Yan Zhang
Yanchun Zhang

Chairs of Special Sessions

Chee Seng Chan
Gea-Jae Joo
Dong-Kyun Kim
Naoyuki Kubota
Kwong Sak Leung
Minho Lee
Yiwen Liang
Bob McKay

Naoki Mori
Xuan Hoai Nguyen
Seiichi Ozawa
Napoleon Reyes
Peter Whigham
Man Leung Wong
Fei Xia
Ingrid Zukerman

Members

Hussein Abbass
Siti Norul Huda Sheikh Abdullah
David Albrecht
Sang-Woo Ban
Mike Barley
Andre Barczak
Laxmidhar Behera
Ghassau Beydoun
Hung Bui
Robin Burke
Longbing Cao
Phoebe Y-P Chen
Songcan Chen
Zheng Chen
Shu-Ching Chen
Sung-Bae Cho
Sungzoon Cho
Key-Sun Choi
Ho-Jin Choi
Seungjin Choi
Jirapun Daengdej
Minh B. Do
Anh Duc Duong
Simon Egerton
Farshad Fahimi
Mohamad Faizal Ahmad Fauzi
Christian Freksa
Dragan Gamberger
Ji Gao

Sharon XiaoYing Gao
Yang Gao
Xin Geng
Guido Governatori
Fikret Gürgen
James Harland
Takashi Hashimoto
Jin-Hyuk Hong
Xiangji Huang
Van Nam Huynh
Hisashi Handa
Renato Ianella
Sanjay Jain
Long Jiang
Geun Sik Jo
Ken Kaneiwa
Byeong Ho Kang
Kyung-Joong Kim
Minkoo Kim
Min Young Kim
In-Cheol Kim
Masahiro Kimura
Alistair Knott
Kazunori Komata
Peep Küngas
Satoshi Kurihara
Sadao Kurohashi
Young-Bin Kwon
Olivia Kwong

Weng Kin Lai
 Eun-Seok Lee
 Yun-Jung Lee
 Yuefeng Li
 Chun-Hung Li
 Zhoujun Li
 Xue-Long Li
 Fangzhen Lin
 Ho-fung Leung
 Chen Change Loy
 Qin Lu
 Michael J. Maher
 Erie Martin
 Enrique Frias Martinez
 Yuji Matsumoto
 Christopher Messom
 Antonija Mitrovic
 Riichiro Mizoguchi
 Diego Molla-Aliod
 Sanguk Noh
 Lars Nolle
 Manabu Okumura
 Seiichi Ozawa
 Seungsoo Park
 Hyeyoung Park
 Seong-Bae Park
 Jose M Pena
 Anton Satria Prabuwo
 Hiok Chai Quek
 Debbie Richards
 Kazumi Saito
 YingPeng Sang
 Rolf Schwitter
 Rudy Setiono
 Yidong Shen
 Akira Shimazu
 Tony Smith
 Safeullah Soomro
 Maosong Sun
 Shahrel Azmin Sundi
 Ho Ha Sung
 Wing Kin Sung
 Nikom Suvonvorn
 An Hwee Tan

David Taniar
 Ban Tao
 Thanaruk Theeramunkong
 Kai Ming Ting
 Cao Son Tran
 Toby Walsh
 Guoyin Wang
 Lipo Wang
 Ian Watson
 Wayne Wobcke
 Xindong Wu
 Mingrui Wu
 Xiangyang Xue
 Seiji Yamada
 Koiehiro Yamauchi
 Tomohiro Yamaguchi
 Jihoon Yang
 Roland H.C Yap
 Dit-Yan Yeung
 Jian Yu
 Lei Yu
 Shipeng Yu
 Pong Chi Yuen
 Bo Zhang
 Changshui Zhang
 Daoqiang Zhang
 Dongmo Zhang
 Du Zhang
 Jumping Zhang
 Liqing Zhang
 Min-Ling Zhang
 Mingyi Zhang
 Shichao Zhang
 Xuegong Zhang
 Zili Zhang
 Tiejun Zhao
 Yanchang Zhao
 Aoying Zhou
 Guodong Zhou
 Yan Zhou
 Xinquan Zhu
 Chengqing Zong

Additional Reviewers

Xiongcai Cai

Yi Cai

Matt Duckham

Tan Yee Fan

Faisal Farooq

Bjoern Gottfried

Yang Sok Kim

Chavalit Likitvivanavong

Bo Liu

Jianbing Ma

H.T. Ng

D. Nguyen

Minh Le Nguyen

Florian Roehrbein

Holger Schulteis

Xiaowei Shao

Pengyi Yang

Dengji Zhao

Organized by



Sponsored by



Table of Contents

Keynotes

Towards Artificial Systems: What Can We Learn from Human Perception?	1
<i>Hemrich H. Bülthoff and Lewis L. Chuang</i>	
Exploiting Macro and Micro Relations toward Web Intelligence	4
<i>Mitsuru Ishizuka</i>	
Speech Recognition for Mobile Devices at Google	8
<i>Mike Schuster</i>	
Symmetry within and between Solutions	11
<i>Toby Walsh</i>	

Regular Papers

Belief Change in OCF-Based Networks in Presence of Sequences of Observations and Interventions: Application to Alert Correlation	14
<i>Salem Benferhat and Karim Tabia</i>	
A Context-Sensitive Manifold Ranking Approach to Query-Focused Multi-document Summarization	27
<i>Xiaoyan Cai and Wenjie Li</i>	
A Novel Approach to Compute Similarities and Its Application to Item Recommendation	39
<i>Christian Desrosiers and George Karypis</i>	
Multiobjective Optimization Approach for Named Entity Recognition	52
<i>Asif Ekbal, Sriparna Saha, and Christoph S. Garbe</i>	
Local Search for Stable Marriage Problems with Ties and Incomplete Lists	64
<i>Mirco Gelain, Maria Silvia Pini, Francesca Rossi, Kristen Brent Venable, and Toby Walsh</i>	
Layered Hypernetwork Models for Cross-Modal Associative Text and Image Keyword Generation in Multimodal Information Retrieval	76
<i>Jung-Woo Ha, Byoung-Hee Kim, Bado Lee, and Byoung-Tak Zhang</i>	
Visual Query Expansion via Incremental Hypernetwork Models of Image and Text	88
<i>Min-Oh Heo, Myunggu Kang, and Byoung-Tak Zhang</i>	

Sampling Bias in Estimation of Distribution Algorithms for Genetic Programming Using Prototype Trees	100
<i>Kangil Kim, Bob (R.I.) McKay, and Dharani Punithan</i>	
Identification of Non-referential Zero Pronouns for Korean-English Machine Translation.....	112
<i>Kye-Sung Kim, Seong-Bae Park, Hyun-Je Song, Se Young Park, and Sang-Jo Lee</i>	
Identifying Idiomatic Expressions Using Phrase Alignments in Bilingual Parallel Corpus	123
<i>Hyoung-Gyu Lee, Min-Jeong Kim, Gumwon Hong, Sang-Bum Kim, Young-Sook Hwang, and Hac-Chang Rim</i>	
Generating an Efficient Sensor Network Program by Partial Deduction.....	134
<i>Li Li and Kerry Taylor</i>	
Conditional Localization and Mapping Using Stereo Camera.....	146
<i>Jigang Liu, Maylor Karhang Leung, and Daming Shi</i>	
A Unified Approach for Extracting Multiple News Attributes from News Pages	157
<i>Wei Liu, Hualiang Yan, Jianwu Yang, and Jianguo Xiao</i>	
A Method for Mobile User Profile and Reasoning	170
<i>Wei Liu and Zhoujun Li</i>	
Evaluating Importance of Websites on News Topics	182
<i>Yajie Miao, Chunping Li, Liu Yang, Lili Zhao, and Ming Gu</i>	
A Statistical Interestingness Measures for XML Based Association Rules	194
<i>Izwan Nizal Mohd Shahrance, Fedja Hadzic, and Tharam S. Dillon</i>	
Toward Improving Re-coloring Based Clustering with Graph b-Coloring	206
<i>Hiroki Ogino and Tetsuya Yoshida</i>	
Semi-supervised Constrained Clustering: An Expert-Guided Data Analysis Methodology	219
<i>Vid Podpečan, Miha Grčar, and Nada Lavrač</i>	
Partial Weighted MaxSAT for Optimal Planning	231
<i>Nathan Robinson, Charles Gretton, Duc Nghia Pham, and Abdul Sattar</i>	
Efficient Estimation of Cumulative Influence for Multiple Activation Information Diffusion Model with Continuous Time Delay	244
<i>Kazumi Saito, Masahiro Kimura, Kouzon Ohara, and Hiroshi Motoda</i>	

Two Natural Heuristics for 3D Packing with Practical Loading Constraints	256
<i>Lei Wang, Songshan Guo, Shi Chen, Wenbin Zhu, and Andrew Lim</i>	
Geometric Median-Shift over Riemannian Manifolds	268
<i>Yang Wang and Xiaodi Huang</i>	
Multi-manifold Clustering	280
<i>Yong Wang, Yuan Jiang, Yi Wu, and Zhi-Hua Zhou</i>	
Exploiting Word Cluster Information for Unsupervised Feature Selection	292
<i>Qingyao Wu, Yunming Ye, Michael Ng, Hanjing Su, and Joshua Huang</i>	
Sparse Representation: Extract Adaptive Neighborhood for Multilabel Classification	304
<i>Shuo Xiang, Songcan Chen, and Lishan Qiao</i>	
Time-Sensitive Feature Mining for Temporal Sequence Classification	315
<i>Yong Yang, Longbing Cao, and Li Liu</i>	
Learning Automaton Based On-Line Discovery and Tracking of Spatio-temporal Event Patterns	327
<i>Anis Yazidi, Ole-Christoffer Granmo, Min Lin, Xifeng Wen, B. John Oommen, Martin Gerdes, and Frank Reichert</i>	
A Graph Model for Clustering Based on Mutual Information	339
<i>Tetsuya Yoshida</i>	
Shill Bidder Detection for Online Auctions	351
<i>Tsuyoshi Yoshida and Hayato Ohwada</i>	
Mining Hot Clusters of Similar Anomalies for System Management	359
<i>Dapeng Zhang, Fen Lin, Zhongzhi Shi, and Heqing Huang</i>	
A Stratified Model for Short-Term Prediction of Time Series	372
<i>Yihao Zhang, Mehmet A. Orgun, Rohan Baxter, and Weiqiang Liu</i>	
Using ASP to Improve the Information Reuse in Mechanical Assembly Sequence Planning	384
<i>Lingzhong Zhao, Xuesong Wang, Junyan Qian, and Tianlong Gu</i>	

Special Session Papers

Manifold Alpha-Integration	397
<i>Heeyoul Choi, Seungjin Choi, Anup Katake, Yoonsop Kang, and Yoonsuck Choe</i>	

Ranking Entities Similar to an Entity for a Given Relationship	409
<i>Yong-Jin Han, Seong-Bae Park, Sang-Jo Lee, Se Young Park, and Kweon-Yang Kim</i>	
Anomaly Detection over Spatiotemporal Object Using Adaptive Piecewise Model	421
<i>Fazli Hanapiah, Ahmed A. Al-Obaidi, and Chee Seng Chan</i>	
Experimental Analysis of the Effect of Dimensionality Reduction on Instance-Based Policy Optimization	433
<i>Hisashi Handa</i>	
A Real-Time Personal Authentication System with Selective Attention and Incremental Learning Mechanism in Feature Extraction and Classifier	445
<i>Young-Min Jang, Seiichi Ozawa, and Minho Lee</i>	
An Efficient Face Recognition through Combining Local Features and Statistical Feature Extraction	456
<i>Donghyun Kim and Hyeyoung Park</i>	
Parameter Learning in Bayesian Network Using Semantic Constraints of Conversational Feedback	467
<i>Seung-Hyun Lee, Sungsoo Lim, and Sung-Bae Cho</i>	
Keystroke Dynamics Extraction by Independent Component Analysis and Bio-matrix for User Authentication	477
<i>Thanh Tran Nguyen, Thai Hoang Le, and Bae Hoai Le</i>	
A Fast Incremental Kernel Principal Component Analysis for Online Feature Extraction	487
<i>Seiichi Ozawa, Yohei Takeuchi, and Shigeo Abe</i>	
Colour Object Classification Using the Fusion of Visible and Near-Infrared Spectra	498
<i>Heesang Shin, Napoleon H. Reyes, Andre L. Barezak, and Chee Seng Chan</i>	
An Adaptive Bidding Strategy for Combinatorial Auction-Based Resource Allocation in Dynamic Markets	510
<i>Xin Sui and Ho-fung Leung</i>	
Online Self-reorganizing Neuro-fuzzy Reasoning in Interval-Forecasting for Financial Time-Series	523
<i>Javan Tan and Chai Quek</i>	
An Evolving Type-2 Neural Fuzzy Inference System	535
<i>Sau Wai Tung, Chai Quek, and Cuntai Guan</i>	

Human Augmented Cognition Based on Integration of Visual and Auditory Information	547
<i>Woong Jae Won, Wono Lee, Sang-Woo Ban, Minook Kim, Hyung-Min Park, and Minho Lee</i>	
Steady-State Genetic Algorithms for Growing Topological Mapping and Localization	558
<i>Jinseok Woo, Naoyuki Kubota, and Beom-Hee Lee</i>	
Incremental Model Selection and Ensemble Prediction under Virtual Concept Drifting Environments	570
<i>Koichiro Yamauchi</i>	
Short Papers	
Multi-dimensional Data Inspection for Supervised Classification with Eigen Transformation Classification Trees	583
<i>Steven De Bruyne and Frank Plaetria</i>	
An Optimised Algorithm to Tackle the Model Explosion Problem in CTL Model Update	589
<i>Yulin Ding and David Hemer</i>	
Exploiting Symmetry in Relational Similarity for Ranking Relational Search Results	595
<i>Tomokazu Goto, Nguyen Tuan Duc, Danushka Bollegala, and Mitsuru Ishizuka</i>	
Brain-Inspired Evolving Neuro-Fuzzy System for Financial Forecasting and Trading of the S&P500 Index	601
<i>Weng Luen Ho, Whye Loon Tung, and Chai Quek</i>	
Bargain over Joint Plans	608
<i>Wei Huang, Dongmo Zhang, Yan Zhang, and Laurent Perrussel</i>	
Point-Based Bounded Policy Iteration for Decentralized POMDPs	614
<i>Youngwook Kim and Kee-Eung Kim</i>	
Chinese Named Entity Recognition Based on Hierarchical Hybrid Model	620
<i>Zihua Liao, Zili Zhang, and Yang Liu</i>	
Text Disambiguation Using Support Vector Machine: An Initial Study	625
<i>Doan Nguyen and Du Zhang</i>	
Diacritics Restoration in Vietnamese: Letter Based vs. Syllable Based Model	631
<i>Kiem-Hieu Nguyen and Cheol-Young Ock</i>	

Tag Quality Feedback: A Framework for Quantitative and Qualitative Feedback on Tags of Social Web	637
<i>Tae-Gil Noh, Jae-Kul Lee, Seong-Bae Park, Se Young Park, Sang-Jo Lee, and Kweon-Yang Kim</i>	
Semantic Networks of Mobile Life-Log for Associative Search Based on Activity Theory	643
<i>Keunhyun Oh and Sung-Bae Cho</i>	
Three-Subagent Adapting Architecture for Fighting Videogames	649
<i>Simón E. Ortiz B., Koichi Moriyama, Ken-ichi Fukui, Satoshi Kurihara, and Masayuki Numao</i>	
Incremental Learning via Exceptions for Agents and Humans: Evaluating KR Comprehensibility and Usability	655
<i>Debbie Richards and Meredith Taylor</i>	
Exploiting Comparable Corpora for Cross-Language Information Retrieval	662
<i>Fatiha Sadat</i>	
Local PCA Regression for Missing Data Estimation in Telecommunication Dataset	668
<i>T. Sato, B.Q. Huang, Y. Huang, and M.-T. Kechedi</i>	
An Influence Diagram Approach for Multiagent Time-Critical Dynamic Decision Modeling	674
<i>Le Sun, Yifeng Zeng, and Yanping Xiang</i>	
Active Learning for Sequence Labelling with Probability Re-estimation	681
<i>Dittaya Wanvarie, Hiroya Takamura, and Manabu Okumura</i>	
Locally Centralizing Samples for Nearest Neighbors	687
<i>Guihua Wen, Si Wen, Jun Wen, and Lijun Jiang</i>	
Gait Planning Research for Biped Robot with Heterogeneous Legs	693
<i>Jun Xiao, Xing Song, Jie Su, and Xinhe Xu</i>	
Computer-Aided Diagnosis of Alzheimers Disease Using Multiple Features with Artificial Neural Network	699
<i>Shih-Ting Yang, Jiann-Der Lee, Chung-Hsien Huang, Jiun-Jie Wang, Wen-Chuin Hsu, and Yau-Yau Wai</i>	
A Hierarchical Multiple Recognizer for Robust Speech Understanding. . .	706
<i>Takahiko Yokoyama, Kazutaka Shimada, and Tsutomu Endo</i>	
Author Index	713

Towards Artificial Systems: What Can We Learn from Human Perception?

Heinrich H. Bülthoff^{1,2} and Lewis L. Chuang¹

¹Max Planck Institute for Biological Cybernetics, Spemannstraße 38,
72076 Tübingen, Germany

²Department of Brain and Cognitive Engineering, Korea University, Anam-dong,
Seongbuk-gu, Seoul, 136-713, Korea

{Heinrich.Buelthoff,Lewis.Chuang}@Springer.com

Abstract. Research in learning algorithms and sensor hardware has led to rapid advances in artificial systems over the past decade. However, their performance continues to fall short of the efficiency and versatility of human behavior. In many ways, a deeper understanding of how human perceptual systems process and act upon physical sensory information can contribute to the development of better artificial systems. In the presented research, we highlight how the latest tools in computer vision, computer graphics, and virtual reality technology can be used to systematically understand the factors that determine how humans perform in realistic scenarios of complex task-solving.

Keywords: perception, object recognition, face recognition, eye-movement, human-machine interfaces, virtual reality, biological cybernetics.

The methods by which we process sensory information and act upon it comprise a versatile control system. We are capable of carrying out a multitude of complex operations, in spite of obvious limitations in our biological “hardware”. These capabilities include our ability to expertly learn and identify objects and people by effectively navigating our eyes and body movements in our visual environment. This talk will present the research perspective of the Biological Cybernetics labs at the Max Planck Institute, Tübingen and the Department of Brain and Cognitive Engineering, Korea University. Key examples will be drawn from our research on face recognition, the relevance of dynamic information and active vision; in order to convey how perceptual research can contribute towards the development of better artificial systems.

To begin, our prodigious ability to learn and remember recently encountered faces – even from only a few instances – reflects a multi-purpose pattern recognition system that few artificial systems can rival, even with the availability of 3D range data. Unintuitively, this perceptual expertise relies on fewer, rather than more, facial features than state-of-the-art face-recognition algorithms typically process. Our visual field of high acuity is extremely limited ($\sim 2^\circ$) and experimental studies indicate that we have an obvious preference for selectively fixating the eyes and noses of faces that we inspect [1]. These facial features inhabit a narrow bandwidth of spatial frequencies (8 to 16 cycles per face), that face-processing competencies are specialized for [2]. Therefore, perceptual expertise appears to result from featural selectivity, wherein

sparse coding by a dedicated system results in expert discrimination. The application of the same principles in artificial systems holds the promise of improving automatic recognition performance.

Self-motion as well as moving objects in our environment dictate that we have to deal with a visual input that is constantly changing. Automated recognition systems would often consider this variability to be a computational hindrance that disrupts the stable retrieval of recognizable object features. Nonetheless, human recognition performance on objects [3] and faces [4] is better served by moving rather than static stimuli. Understanding why this is so, could allow artificial recognition systems to function equally well in dynamic environments. First, dynamic presentations present the opportunity for associative learning between familiar object views, which could result in object representations that are robust to variations in pose [5, 6]. Furthermore, dynamic presentations could allow the perceptual system to assess the stability of different object features, according to how they tend to appear and disappear over rigid rotations. This could offer a computationally cheap method for determining the minimal set of object views that would be sufficient for robust recognition [7, 8]. Finally, characteristic motion properties (e.g., trajectories, velocity profile) could even serve as an additional class of features to complement a traditional reliance on image and shape features by automated recognition systems [9, 10].

Purposeful gaze behavior indicates a perceptual system that is not only capable of processing information, but proficient in seeking out information, too. We are capable of extracting a scene's gist within the first few hundred milliseconds of encountering it [11]. In turn, this information directs movement of our eyes and head for the joint purpose of fixating information-rich regions across a large field of view [12]. In addition, we use our hands to explore and manipulate objects so as to access task-relevant information for object learning or recognition [13, 14, 15]. Careful observations of how we interact with our environments can identify behavioral primitives that could be modeled and incorporated into artificial systems as functional (and re-usable) components [16]. Furthermore, understanding how eye and body movements naturally coordinate can allow us improve the usability of artificial systems [17].

This perspective of the perceptual system as an active control system continues to be insightful at a higher level, when we consider the human operator as a controller component in dynamic machine systems. Take, for example, a pilot who has to simultaneously process visual and vestibular information, in order to control helicopter stability. Using motion platforms and immersive graphics, it is possible to systematically identify the input parameters that are directly relevant to a pilot's task performance and thus, derive a functional relationship between perceptual inputs and performance output [18]. Such research is fundamental for the development of virtual environments that are perceptually realistic. This is especially important when designing artificial systems (e.g., flight simulators) that are intended to prepare novices for physically dangerous situations that are not easily replicable in the real world [19].

Until now, we have discussed how findings from perceptual research can contribute towards improving artificial systems. However, the growing prevalence of these systems in our daily environs raises an imperative to go beyond this goal. It is crucial to consider how perceptual and artificial systems may be integrated into a coherent whole by considering the "human-in-the-loop". Doing so will lead towards a new generation of autonomous systems that will not merely mimic our perceptual competencies, but will be able to cooperate with and augment our natural capabilities.

References

1. Armann, R., Bühlhoff, I.: Gaze Behavior in Face Comparison: The Role of Sex, Task, and Symmetry. *Attention, Perception & Psychophysics* 71, 1107–1126 (2009)
2. Keil, M.S.: “I Look in Your Eyes, Honey”: Internal Face Features Induce Spatial Frequency Preference for Human Face Processing. *PLOS Computational Biology* 5, 1–13 (2009)
3. Chuang, L., Vuong, Q.C., Thornton, I.M., Bühlhoff, H.H.: Recognizing novel deforming objects. *Visual Cognition* 14, 85–88 (2006)
4. Knappmeyer, B., Thornton, I.M., Bühlhoff, H.H.: The use of facial motion and facial form during the processing of identity. *Vision Research* 43, 1921–1936 (2003)
5. Wallis, G.M., Bühlhoff, H.H.: Effects of Temporal Association on Recognition Memory. *Proceedings of the National Academy of Sciences of the United States of America* 98, 4800–4804 (2001)
6. Wallis, G., Backus, B.T., Langer, M., Huebner, G., Bühlhoff, H.H.: Learning Illumination- and Orientation-Invariant Representations of Objects Through Temporal Association. *Journal of Vision* 9, 1–8 (2009)
7. Wallraven, C., Bühlhoff, H.H.: Automatic Acquisition of Exemplar-Based Representations for Recognition from Image Sequences. In: *IEEE Conference on Computer Vision and Pattern Recognition* (2001)
8. Bühlhoff, H.H., Wallraven, C., Graf, A.: View-Based Dynamic Object Recognition Based on Human Perception. In: *Proceedings of the 16th International Conference on Pattern Recognition*, pp. 768–776 (2002)
9. Newell, F.N., Wallraven, C., Huber, S.: The role of characteristic motion on object categorization. *Journal of Vision* 4, 118–129 (2004)
10. Curio, C., Bühlhoff, H.H., Giese, M.A.: *Dynamic Faces: Insights from Experiments and Computation*. MIT Press, Cambridge (2010)
11. Vogel, J.A., Schwaninger, A., Wallraven, C., Bühlhoff, H.H.: Categorization of natural scenes: Local versus Global Information and the Role of Color. *ACM Transactions on Applied Perception* 4, 1–21 (2010)
12. Chuang, L.L., Bieg, H.-J., Fleming, R.W., Bühlhoff, H.H.: Measuring Unrestrained Gaze on Wall-Sized Displays. In: *Proceedings of the 28th European Conference on Cognitive Ergonomics* (2010)
13. Chuang, L.L., Vuong, Q.C., Thornton, I.M., Bühlhoff, H.H.: Human Observers Use Personal Exploration Patterns in Novel Object Recognition. *Perception* 36(Suppl.), 49 (2007)
14. Blanz, V., Tarr, M.J., Bühlhoff, H.H.: What Object Attributes Determine Canonical Views? *Perception* 28, 575–600 (1999)
15. Gaßbert, N., Wallraven, C.: Perceptual Representations of Parametrically-Defined and Natural Objects Comparing Vision and Haptics. In: *Proceedings of the Haptics Symposium*, pp. 1–8 (2010)
16. Sprague, N., Ballard, D., Robinson, A.: Modeling Embodied Visual Behaviors. *Transactions on Applied Perception* 4, 1–26 (2007)
17. Bieg, H.-J., Chuang, L.L., Fleming, R.W., Reiterer, H., Bühlhoff, H.H.: Eye and Pointer Coordination in Search and Selection Tasks. In: *Proceedings of the Symposium on Eye Tracking Research and Applications*, pp. 89–92 (2010)
18. Nieuwenhuizen, F.M., Zaal, P.M.T., Teufel, H., Mulder, M., Bühlhoff, H.H.: The Effect of Simulator Motion on Pilot Control Behaviour for Agile and Inert Helicopter Dynamics. In: *35th European Rotorcraft Forum*, pp. 1–13 (2009)
19. Niccolini, M.L., Pollini, L., Innocenti, M., Robuffo Giordano, P., Teufel, H., Bühlhoff, H.H.: Towards Real-Time Aircraft Simulation with the MPI Motion Simulator. In: *Proceedings of the 2009 AIAA Modeling and Simulation Technologies Conference*, pp. 1–10 (2009)

Exploiting Macro and Micro Relations toward Web Intelligence

Mitsuru Ishizuka

School of Information Science and Technology
University of Tokyo
7-3-1, Hongo, Bunkyo-ku, Tokyo 113-8656, Japan
ishizuka@i.u-tokyo.ac.jp

Keywords: Relation extraction/mining, relational similarity, relational search, semantic computing, Web intelligence.

Relations are basic elements for representing knowledge, such as in semantic network, logic and others. In Web intelligence research, the extraction or mining of meaningful knowledge and the utilization of the knowledge for intelligent services are key issues. In this talk, I will present some of our researches related to these issues, ranging from macro relations to micro ones. Here we mostly use Web texts, and the use of their huge data though a search engine becomes a key function together with text analysis.

The first topic concerns with the extraction of human-human and company-company relations from the Web [1-14]. Relation types between two entities are also extracted here. An open Web service based on this function has been operated in Japan by a company. One technology related to this one is namesake disambiguation [15-17].

Wikipedia is a good reliable source for wide knowledge, unlike other Web information. In order to extract the knowledge or data from Wikipedia in the form that computers can understand and manipulate, several attempts including ours [18-23] have been carried out, typically to extract triplets such as (entity, attribute, value).

After we worked on computing similarity between two words based on the distributional hypothesis [24, 25], we have been interested in computing similarity between two word pairs (or two entity pairs) [26-28]. Like in the previous case, we are mainly utilizing distributional hypothesis, and have invented an efficient clustering method for dealing with several tens of thousands of lexical patterns. Based on this mechanism, we have implemented a latent relational search engine, which accepts two entity pairs with one missing component such as {(Tokyo, Japan), (?, France)} as a query, and produces an answer such as (? = Paris) with its evidence. As an extension of this mechanism, we recently invented an efficient co-clustering method, which works well to find arbitrary existing relations between two nouns in sentences [29]. This problem setting is called open information extraction (open IE).

The final topic of the talk is Concept Description Language (CDL), which has been designed to serve as a common language for representing concept meaning expressed in natural language texts [30-32]. Unlike Semantic Web which provides machine-readable meta-data in the form of RDF, CDL aims to encode the meaning of the whole texts in a machine-understandable form. The basic representation element in CDL is micro relations existing between entities in the text; 44 relation types are defined. CDL has been discussed in a W3C incubator group for international standardization since 2007. It is intended to be a basis of semantic computing in next generation, and also become a medium for overcoming language barrier in the world. Current issues of CDL are, among others, an easy semi-automatic way of converting natural language texts into the CDL description, and an effective mechanism of semantic retrieval on the CDL database.

References

1. Matsuo, Y., Tomobe, H., Hasida, K., Ishizuka, M.: Mining Social Network of Conference Participants from the Web. In: Proc. 2003 IEEE/WIC Int'l Conf. on Web Intelligence (WI 2003), Halifax, Canada (2003)
2. Mori, J., Matsuo, Y., Ishizuka, M., Faltings, B.: Keyword Extraction from the Web for FOAF Metadata. In: Proc. 1st Workshop on Friend of a Friend, Social Networking and the Semantic Web, Galway, Ireland, pp. 1-8 (2004)
3. Mori, J., Sugiyama, T., Matsuo, Y., Tomobe, H., Ishizuka, M.: Real-world Oriented Information Sharing using Social Network. In: Proc. 25th Int'l Sunbelt Social Network Conf, SUNBELT XXV (2005)
4. Mori, J., Matsuo, Y., Hashida, K., Ishizuka, M.: Web Mining Approach for a User-centered Semantic Web. In: Proc. Int'l Workshop on User Aspects on the Semantic Web in 2nd European Semantic Web Conf. (ESWC 2005), Heraklion, Greece, pp. 177-187 (2005)
5. Matsuo, Y., Mori, J., Hamasaki, M., Ishida, K., Nishimura, T., Takeda, H., Hasida, K., Ishizuka, M.: POLYHONET: An Advanced Social Network Extraction System from the Web. In: Proc. 15th World Wide Web Conf. (WWW 2006), Edinburgh, UK (2006) (CD-ROM)
6. Matsuo, Y., Hamasaki, M., Nakamura, Y., Nishimura, T., Hasida, K., Takeda, H., Mori, J., Bollegala, D., Ishizuka, M.: Spinning Multiple Social Networks for Semantic Web. In: Proc. 21st National Conf. on Artificial Intelligence (AAAI 2006), Boston, MA, USA, pp. 1381-1387 (2006)
7. Mori, J., Tsujishita, T., Matsuo, Y., Ishizuka, M.: Extracting Relations in Social Networks from the Web Using Similarity Between Collective Contexts. In: Cruz, I., Decker, S., Allemang, D., Preist, C., Schwabe, D., Mika, P., Uschold, M., Aroyo, L.M. (eds.) ISWC 2006. LNCS, vol. 4273, pp. 487-500. Springer, Heidelberg (2006)
8. Mori, J., Matsuo, Y., Ishizuka, M.: Extracting Keyphrases to Represent Relations in Social Networks from Web. In: Proc. 20th Int'l Joint Conf. on Artificial Intelligence (IJCAI 2007), Hyderabad, India, pp. 2820-2825 (2007)
9. Matsuo, Y., Mori, J., Ishizuka, M.: Social Network Mining from the Web. In: Poncelet, P., Teisseire, M., Masegla, F. (eds.) Data Mining Patterns - New Methods and Applications, ch. VII, pp. 149-175. Information Science Reference (2007)
10. Matsuo, Y., Mori, J., Hamasaki, M., Nishimura, T., Takeda, H., Hasida, K., Ishizuka, M.: POLYHONET: An Advanced Social Network Extraction System from the Web. *Journal of Web Semantics* 5(4), 262-278 (2007)

11. Jin, Y., Matsuo, Y., Ishizuka, M.: Extracting a Social Network among Entities by Web Mining. In: Proc. ISWC 2006 Workshop on Web Content Mining with Human Language Technologies, Athens, GA, USA, 10 p. (2006)
12. Jin, Y., Matsuo, Y., Ishizuka, M.: Extracting Social Networks among Various Entities on the Web. In: Franconi, E., Kifer, M., May, W. (eds.) ESWC 2007. LNCS, vol. 4519, pp. 251–266. Springer, Heidelberg (2007)
13. Jin, Y., Ishizuka, M., Matsuo, Y.: Extracting Inter-firm Networks from the World Wide Web using a General-purpose Search Engine. *Information Review* 32(2), 196–210 (2008)
14. Jin, Y., Matsuo, Y., Ishizuka, M.: Ranking Companies Based on Multiple Social Networks Mined from the Web. In: Kang, K. (ed.) E-commerce, INTECH, ch. 6, pp. 75–98 (2010)
15. Bollegala, D., Matsuo, Y., Ishizuka, M.: Extracting Key Phrases to Disambiguate Personal Names on the Web. In: Gelbukh, A. (ed.) CICLing 2006. LNCS, vol. 3878, pp. 223–234. Springer, Heidelberg (2006)
16. Bollegala, D., Matsuo, Y., Ishizuka, M.: Extracting Key Phrases to Disambiguate Personal Name Queries in Web Search. In: Proc. of the Workshop “How can Computational Linguistics Improve Information Retrieval?” at the Joint 21st Int’l Conf. on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics (COLING-ACL 2006), Sydney, Australia, pp. 17–24 (2006)
17. Bollegala, D., Matsuo, Y., Ishizuka, M.: Disambiguating Personal Names on the Web using Automatically Extracted Key Phrases. In: Proc. European Conf. on Artificial Intelligence (ECAI 2006), Trento, Italy, pp. 553–557 (2006)
18. Nguyen, D.P.T., Matsuo, Y., Ishizuka, M.: Exploiting Syntactic and Semantic Information for Relation Extraction from Wikipedia. In: Proc. IJCAI 2007 Workshop on Text-Mining and Link-Analysis (TextLink 2007), Hyderabad, India, 11 p. (2007) (CD-ROM)
19. Nguyen, D.P.T., Matsuo, Y., Ishizuka, M.: Subtree Mining for Relation Extraction from Wikipedia. In: Companion Volume of Proc. of the Main Conf. of Human Language Technologies 2007: The Conf. of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL 2007), Rochester, New York, pp. 125–128 (2007)
20. Nguyen, D.P.T., Matsuo, Y., Ishizuka, M.: Relation Extraction from Wikipedia Using Subtree Mining. In: Proc. 22nd Conf. on Artificial Intelligence (AAAI 2007), pp. 1414–1420 (2007)
21. Watanabe, K., Bollegala, D., Matsuo, Y., Ishizuka, M.: A Two-Step Approach to Extracting Attributes for People on the Web. In: Proc. WWW 2009 2nd Web People Search Evaluation Workshop (WEPS 2009), Madrid, Spain, 6 p. (2009)
22. Yan, Y., Okazaki, N., Matsuo, Y., Yang, Z., Ishizuka, M.: Unsupervised Relation Extraction by Mining Wikipedia Texts Using Information from the Web. In: Proc. of Joint Conf. of 47th Annual Meeting of the Association for Computational Linguistics and 4th Int’l Joint Conf. on Natural Language Processing of the Asian Federation of Natural Language Processing (ACL-IJCNLP 2009), Singapore, pp. 1021–1029 (2009)
23. Yan, Y., Li, H., Matsuo, Y., Ishizuka, M.: Multi-view Bootstrapping for Relation Extraction by Exploiting Web Features and Linguistic Features. In: Gelbukh, A. (ed.) CICLing 2010. LNCS, vol. 6008, pp. 525–536. Springer, Heidelberg (2010)
24. Bollegala, D., Matsuo, Y., Ishizuka, M.: Measuring Semantic Similarity between Words Using Web Search Engines. In: Proc. 16th Int’l World Wide Web Conf. (WWW 2007), Banff, Canada, pp. 757–766 (2007)
25. Bollegala, D., Matsuo, Y., Ishizuka, M.: WWW sits the SAT: Measuring Relational Similarity from the Web. In: Proc. 18th European Conf. on Artificial Intelligence (ECAI 2008), Patras, Greece, pp. 333–337 (2008)

26. Bollegala, D., Matsuo, Y., Ishizuka, M.: Measuring the Similarity between Implicit Semantic Relations using Web Search Engines. In: Proc. 2009 Second ACM Int'l Conf. on Web Search and Data Mining (WSDM 2009), Barcelona, Spain, pp. 104–113 (2009) (CD-ROM)
27. Bollegala, D., Matsuo, Y., Ishizuka, M.: Measuring the Similarity between Implicit Semantic Relations from the Web. In: Proc. 18th Int'l World Wide Web Conf. (WWW 2009), Madrid, Spain, pp. 651–660 (2009)
28. Bollegala, D., Matsuo, Y., Ishizuka, M.: A Relational Model of Semantic Similarity between Words using Automatically Extracted Lexical Pattern Clusters from the Web. In: Proc. 2009 Conf. on Empirical Methods in Natural Language Processing (EMNLP 2009), Singapore, pp. 803–812 (2009)
29. Bollegala, D., Matsuo, Y., Ishizuka, M.: Relational Duality: Unsupervised Extraction of Semantic Relations between Entities on the Web. In: Proc. 19th Int'l World Wide Web Conf. (WWW 2010), Raleigh, North Carolina, USA, pp. 151–160 (2010)
30. Report of W3C Incubator Group on Common Web Language (2008), <http://www.w3.org/2005/Incubator/cwl/XGR-cwl-20080331/>
31. Yan, Y., Matsuo, Y., Ishizuka, M., Yokoi, T.: Annotating an Extension Layer of Semantic Structure for Natural Language Text. In: Proc. 2nd IEEE Int'l Conf. on Semantic Computing, Santa Clara, CA, USA, pp. 174–181 (2008) (CD-ROM)
32. Yan, Y., Matsuo, Y., Ishizuka, M., Yokoi, T.: Relation Classification for Semantic Structure Annotation of Text. In: Proc. 2008 IEEE/WIC/ACM Int'l Conf. on Web Intelligence (WI 2008), Sydney, Australia, pp. 377–380 (2008) (CD-ROM)

Speech Recognition for Mobile Devices at Google

Mike Schuster

Google Research, 1600 Amphitheatre Pkwy., Mountain View, CA 94043, USA
schuster@google.com

Abstract. We briefly describe here some of the content of a talk to be given at the conference.

1 Introduction

At Google, we focus on making information universally accessible through many channels, including through spoken input. Since the speech group started in 2005 we have developed several successful speech recognition services for the US and for some other countries. In 2006 we launched GOOG-411 in the US, a speech recognition driven directory assistance service which works from any phone. As smartphones like the iPhone, BlackBerry, Nokia s60 platform and phones running the Android operating system like the Nexus One and others becoming more widely used we shifted our efforts to provide speech input for the search engine (Search by Voice) and other applications on these phones. Many recent smartphones have only soft keyboards which can be difficult to type on, especially for longer input words and sentences. Some Asian languages, for example Japanese and Chinese are more difficult to type as the basic number of characters is very high compared to Latin alphabet languages. Spoken input is a natural choice to improve on many of these problems, and more details are discussed in the sections below.

We have also been working on voice mail transcription and YouTube transcription for US English, which are also publically available products in the US, but the focus here will be on speech recognition in the context of mobile devices.

2 GOOG-411

GOOG-411 is Google's speech recognition based directory assistance service operating in the US and Canada [1], [2]. This application uses a toll-free number, 1-800-GOOG-411 (1-800-4664-411). The user is prompted to say city, state and the name of the business s(he) is looking for. Using text-to-speech the service can give address and phone number, or can connect the user directly to the business. As backend information from Google Maps Local is used.

While this is a useful application to search for restaurants, stores etc. it is limited to businesses. Other difficulties with this kind of service include the

necessity of a dialog, relatively expensive operating costs, listing errors in the backend database, and most importantly to not be able to give richer information (as on a smartphone screen) back to the user.

3 Voice Search

In 2008 Google launched Voice Search in the US for several types of smartphones [3]. Voice Search adds simply the ability to speak a search query to the phone instead of having to type it into the browser. The audio is sent to Google servers where it is recognized and the recognition result along with the search result is sent back to the phone. The data goes over the data channel instead of the voice channel which allows higher quality audio transmission and therefore better recognition rates. Our speech recognition technology is relatively standard, below some details.

Front-End and Acoustic Model. For the front-end we use 39-dimensional PLP features with LDA. The acoustic models are ML and MMI trained, triphone decision-tree tied 3-state HMMs with currently up to 10k states total. The state distributions are modeled by 50-300k diagonal covariance Gaussians with STC. We use a time-synchronous finite-state transducer (FST) decoder with Gaussian selection for speedy likelihood calculation.

Dictionary. Our phone set contains between 30 and 100 phones depending on the language. We use between 200k and 1.5M words in the dictionary, which are automatically extracted from the web-based query stream. The pronunciations for these words are mostly generated by an automatic system with special treatment for numbers, abbreviations and other exceptions.

Language Model. As our goal is to recognize search queries we mine our language model data from web-based anonymous search queries. We mostly use 3-grams or 5-grams with Katz backoff trained on months or years of query data. The language models have to be primed appropriately such that the final decoder graphs fit into memory of the servers.

Acoustic Data. To train an initial system we collect roughly 250k of spoken queries using an Android application specifically designed for this purpose [4]. Several hundred speakers read queries off a screen and the corresponding voice samples are recorded. As most queries are spoken without errors we don't have to manually transcribe these queries.

Metrics. We want to optimize user experience. Traditionally speech recognition systems focus on minimizing word error rate. This is also a useful measure for us, but better is a normalized sentence error rate as it doesn't depend as much on the definition of a word. As the metric which approximates user experience best we use WebScore: We send hypothesis and reference to a search backend and

compare the links we get back. Assuming that the reference generates the correct search result this way we know whether the search result for the hypothesis is within the first three results – such that the user can see the correct result on his smartphone screen.

Languages. After US English we launched Voice Search for the UK, Australia and India. Late 2009 Mandarin Chinese [5] and Japanese were added. Foreign languages pose many additional challenges. For example, some Asian languages like Japanese and Chinese don't have spaces between words. For these we wrote a segmenter which optimizes the word definitions maximizing sentence likelihood. Most languages have characters outside the normal ASCII set, in some cases thousands, which complicate automatic pronunciation rules.

Additional Challenges. There are many details which are critical to get right for a good user experience but we cannot discuss here because of space constraints. These include getting the user interface right, optimizing protocols for minimum latency, dealing with special cases like numbers, dates and abbreviations correctly, avoid showing offensive queries and improving the system efficiently after launch using the data coming in.

4 Outlook

For mobile devices speech is an attractive input modality and besides Voice Search we have been working on other features, including moer general Voice Input [6], contact dailing (as launched in the US) and recognition of special phrases to trigger certain applications on the phone. We believe that in the next few years speech input will become more accurate, more accepted and useful enough to help users efficiently access and navigate through information provided through mobile devices.

References

1. Bacchiani, M., Beaufays, F., Schalkwyk, J., Schuster, M., Strobe, B.: Deploying GOOG-411: Early lessons in data, measurement, and testing. In: Proceedings of ICASSP, pp. 5260–5263 (2008)
2. van Heerden, C., Schalkwyk, J., Strobe, B.: Language Modeling for What-with-Where on GOOG-411. In: Proceedings of Interspeech, pp. 991–994 (2009)
3. Schalkwyk, J., Beeferman, D., Beaufays, F., Byrne, B., Chelba, C., Cohen, M., Kamvar, M., Strobe, B.: Google Search by Voice: A case study. In: Weinstein, A. (ed.) *Visions of Speech: Exploring New Voice Apps in Mobile Environments, Call Centers and Clinics*. Springer, Heidelberg (2010) (in Press)
4. Hughes, T., Nakajima, K., Ha, L., Vasu, A., Moreno, P., LeBeau, M.: Building transcribed speech corpora quickly and cheaply for many languages. In: Interspeech (submitted 2010)
5. Shan, J., Wu, G., Hu, Z., Tang, X., Jansche, M., Moreno, P.: Search by Voice in Mandarin Chinese. In: Interspeech (submitted 2010)
6. Ballinger, B., Allauzen, C., Gruenstein, A., Schalkwyk, J.: On-Demand Language Model Interpolation for Mobile Speech Input. In: Interspeech (submitted 2010)

Symmetry within and between Solutions*

Toby Walsh

NICTA and University of NSW, Sydney, Australia
toby.walsh@nicta.com.au

Abstract. Symmetry can be used to help solve many problems. For instance, Einstein's famous 1905 paper ("On the Electrodynamics of Moving Bodies") uses symmetry to help derive the laws of special relativity. In artificial intelligence, symmetry has played an important role in both problem representation and reasoning. I describe recent work on using symmetry to help solve constraint satisfaction problems. Symmetries occur within individual solutions of problems as well as between different solutions of the same problem. Symmetry can also be applied to the constraints in a problem to give new symmetric constraints. Reasoning about symmetry can speed up problem solving, and has led to the discovery of new results in both graph and number theory.

1 Introduction

Symmetry occurs in many combinatorial search problems. For example, in the magic squares problem (prob019 in CSPLib [1]), we have the symmetries that rotate and reflect the square. Eliminating such symmetry from the search space is often critical when trying to solve large instances of a problem. Symmetry can occur both *within* a single solution as well as *between* different solutions of a problem. We can also *apply* symmetry to the constraints in a problem. We focus here on constraint satisfaction problems, though there has been interesting work on symmetry in other types of problems (e.g. planning, and model checking). We summarize recent work appearing in [2,3,4].

2 Symmetry between Solutions

A symmetry σ is a bijection on assignments. Given a set of assignments A and a symmetry σ , we write $\sigma(A)$ for $\{\sigma(a) \mid a \in A\}$. A special type of symmetry, called *solution symmetry* is a symmetry *between* the solutions of a problem. More formally, we say that a problem has the *solution symmetry* σ iff σ of any solution is itself a solution [5].

Running example: *The magic squares problem is to label a n by n square so that the sum of every row, column and diagonal are equal (prob019 in CSPLib [1]). A normal magic square contains the integers 1 to n^2 . We model this with n^2 variables $X_{i,j}$ where $X_{i,j} = k$ iff the i th column and j th row is labelled with the integer k .*

* Supported by the Australian Government's Department of Broadband, Communications and the Digital Economy and the ARC. Thanks to the co-authors of the work summarized here: Marijn Heule, George Katsirelos and Nina Narodytska.

“Lo Shu”, the smallest non-trivial normal magic square has been known for over four thousand years and is an important object in ancient Chinese mathematics:

$$\begin{array}{|c|c|c|} \hline 4 & 9 & 2 \\ \hline 3 & 5 & 7 \\ \hline 8 & 1 & 6 \\ \hline \end{array} \tag{1}$$

The magic squares problem has a number of solution symmetries. For example, consider the symmetry σ_d that reflects a solution in the leading diagonal. This map “Lo Shu” onto a symmetric solution:

$$\begin{array}{|c|c|c|} \hline 6 & 7 & 2 \\ \hline 1 & 5 & 9 \\ \hline 8 & 3 & 4 \\ \hline \end{array} \tag{2}$$

Any other rotation or reflection of the square maps one solution onto another. The 8 symmetries of the square are thus all solution symmetries of this problem. In fact, there are only 8 different magic square of order 3, and all are in the same symmetry class.

One way to factor solution symmetry out of the search space is to post symmetry breaking constraints. See, for instance, [6,7,8,9,10,11,12,13,14]. For example, we can eliminate σ_d by posting a constraint which ensures that the top left corner is smaller than its symmetry, the bottom right corner. This selects (1) and eliminates (2). Symmetry can be used to transform such symmetry breaking constraints [2]. For example, if we apply σ_d to the constraint which ensures that the top left corner is smaller than the bottom right, we get a new symmetry breaking constraints which ensures that the bottom right is smaller than the top left. This selects (2) and eliminates (1).

3 Symmetry within a Solution

Symmetries can also be found within individual solutions of a constraint satisfaction problem. We say that a solution A contains the internal symmetry σ (or equivalently σ is an internal symmetry within this solution) iff $\sigma(A) = A$.

Running example: Consider again “Lo Shu”. This contains an internal symmetry. To see this, consider the solution symmetry σ_{inv} that inverts labels, mapping k onto $n^2 + 1 - k$. This solution symmetry maps “Lo Shu” onto a different (but symmetric) solution. However, if we now apply the solution symmetry σ_{180} that rotates the square 180° , we map back onto the original solution:

$$\begin{array}{|c|c|c|} \hline 4 & 9 & 2 \\ \hline 3 & 5 & 7 \\ \hline 8 & 1 & 6 \\ \hline \end{array} \begin{array}{l} \sigma_{inv} \\ \Rightarrow \\ \Leftarrow \\ \sigma_{180} \end{array} \begin{array}{|c|c|c|} \hline 6 & 1 & 8 \\ \hline 7 & 5 & 3 \\ \hline 2 & 9 & 4 \\ \hline \end{array}$$

Consider the composition of these two symmetries: $\sigma_{inv} \circ \sigma_{180}$. As this maps “Lo Shu” onto itself, the solution “Lo Shu” contains the internal symmetry $\sigma_{inv} \circ \sigma_{180}$.

In general, there is no relationship between the solution symmetries of a problem and the internal symmetries within a solution of that problem. There are solution symmetries of a

problem which are not internal symmetries within any solution of that problem, and vice versa. However, when all solutions of a problem contain the same internal symmetry, we can be sure that this is a solution symmetry of the problem itself. The exploitation of internal symmetries involves two steps: finding internal symmetries, and then restricting search to solutions containing just these internal symmetries. We have explored this idea in two applications where we have been able to extend the state of the art. In the first, we found new lower bound certificates for Van der Waerden numbers. Such numbers are an important concept in Ramsey theory. In the second application, we increased the size of graceful labellings known for a family of graphs. Graceful labelling has practical applications in areas like communication theory. Before our work, the largest double wheel graph that we found graceful labelled in the literature had size 10. Using our method, we constructed the first known labelling for a double wheel of size 24.

References

1. Gent, I., Walsh, T.: CSPLib: a benchmark library for constraints. Technical report APES-09-1999, A shorter version appears in CP-99 (1999)
2. Heule, M., Walsh, T.: Symmetry within solutions. In: Proc. of 24th National Conf. on AI. AAAI, Menlo Park (2010)
3. Katsirelos, G., Walsh, T.: Symmetries of symmetry breaking constraints. In: Proc. of 19th ECAI (2010)
4. Katsirelos, G., Narodytska, N., Walsh, T.: Static constraints for breaking row and column symmetry. In: 16th Int. Conf. on Principles and Practices of Constraint Programming, CP 2010 (2010) (under review)
5. Cohen, D., Jeavons, P., Jefferson, C., Petrie, K., Smith, B.: Symmetry definitions for constraint satisfaction problems. *Constraints* 11, 115–137 (2006)
6. Puget, J.F.: On the satisfiability of symmetrical constrained satisfaction problems. In: Komorowski, J., Raś, Z.W. (eds.) ISMIS 1993. LNCS, vol. 689, pp. 350–361. Springer, Heidelberg (1993)
7. Crawford, J., Ginsberg, M., Luks, G., Roy, A.: Symmetry breaking predicates for search problems. In: Proc. of the 5th Int. Conf. on Knowledge Representation and Reasoning (KR 1996), pp. 148–159 (1996)
8. Flener, P., Frisch, A., Hnich, B., Kiziltan, Z., Miguel, I., Pearson, J., Walsh, T.: Breaking row and column symmetry in matrix models. In: Van Hentenryck, P. (ed.) CP 2002. LNCS, vol. 2470, p. 462. Springer, Heidelberg (2002)
9. Frisch, A., Hnich, B., Kiziltan, Z., Miguel, I., Walsh, T.: Global constraints for lexicographic orderings. In: Van Hentenryck, P. (ed.) CP 2002. LNCS, vol. 2470, p. 93. Springer, Heidelberg (2002)
10. Frisch, A., Hnich, B., Kiziltan, Z., Miguel, I., Walsh, T.: Propagation algorithms for lexicographic ordering constraints. *Artificial Intelligence* 170(10), 803–834 (2006)
11. Walsh, T.: General symmetry breaking constraints. In: Benhamou, F. (ed.) CP 2006. LNCS, vol. 4204, pp. 650–664. Springer, Heidelberg (2006)
12. Walsh, T.: Symmetry breaking using value precedence. In: 17th ECAI (2006)
13. Law, Y.-C., Lee, J., Walsh, T., Yip, J.: Breaking symmetry of interchangeable variables and values. In: Bessière, C. (ed.) CP 2007. LNCS, vol. 4741, pp. 423–437. Springer, Heidelberg (2007)
14. Walsh, T.: Breaking value symmetry. In: Bessière, C. (ed.) CP 2007. LNCS, vol. 4741, pp. 880–887. Springer, Heidelberg (2007)
15. Walsh, T.: Breaking value symmetry. In: Proc. of 22nd National Conf. on AI. AAAI, Menlo Park (2008)

Belief Change in OCF-Based Networks in Presence of Sequences of Observations and Interventions: Application to Alert Correlation

Salem Benferhat and Karim Tabia

CRIL CNRS UMR 8188, Artois University, France
{benferhat,tabia}@cril.univ-artois.fr

Abstract. Ordinal conditional function (OCF) frameworks have been successfully used for modeling belief revision when agents' beliefs are represented in the propositional logic framework. This paper addresses the problem of belief change of graphical representations of uncertain information, called OCF-based networks. In particular, it addresses how to revise OCF-based networks in presence of sequences of observations and interventions. This paper contains three contributions: Firstly, we show that the well-known mutilation and augmentation methods for handling interventions proposed in the framework of probabilistic causal graphs have natural counterparts in OCF causal networks. Secondly, we provide an OCF-based counterpart of an efficient method for handling sequences of interventions and observations by directly performing equivalent transformations on the initial OCF graph. Finally, we highlight the use of OCF-based causal networks on the alert correlation problem in intrusion detection.

Keywords: OCF-based networks, belief change, causal reasoning, alert correlation.

1 Introduction

Among the powerful frameworks for representing uncertain pieces of information, ordinal conditional functions (OCF) [12] is an ordinal setting that has been successfully used for modeling revision of agents' beliefs [4]. OCFs are very useful for representing uncertainty and several works point out their relevance for representing agents' beliefs and defining belief change operators for updating the current beliefs in the light of new information [9]. OCF-based networks (also called *kappa*-networks) [7] are graphical models [8] expressing the beliefs using OCF ranking functions. The graphical component allows an easy and compact representation of influence relationships existing between the domain variables while OCFs allow an easy quantification of belief strengths. OCF-based networks are less demanding than probabilistic networks (where exact probability degrees are needed). In OCF-based networks, belief strengths, called degrees of surprise, may be regarded as order of magnitude probability estimates which makes easier the elicitation of agents' beliefs.

Causality is an important notion in many applications such as diagnosis, explanation, simulation, etc. There are several recent approaches and frameworks addressing causality issues in several areas of artificial intelligence. Among these formalisms, causal graphical models (such as causal Bayesian graphs [8] and possibilistic networks [2]) offer efficient tools for causal ascription. While OCF frameworks have been extensively used for studying default reasoning and belief revision, there are only few works addressing belief change in OCF-based networks while causality issues have not yet been investigated.

Observations are often handled using a simple form of conditioning and the order in which they are reported does not matter. The situation is clearly different in the presence of both interventions and observations. Let us consider an example in the intrusion detection field. Assume that for the network administrator, the most common situation is that the Web server works normally and in case where this latter works abnormally or crashes, it is mostly due to flooding denial of service attacks DoS¹ launched by attackers. Now, if one day, the administrator observes that his server works abnormally, then after this observation, any other external action causing his Web server crash will not change his beliefs regarding the fact that a DoS attack is being undertaken. Consider now the converse situation where just before looking at the alert log file (in order to check whether DoS attacks were detected), we perform a manipulation that crashes the Web server². Then after this intervention, without surprise the administrator observes that his server crashes but he will not change his a priori beliefs concerning the fact that there is no attack which is currently undergoing. Here, an observation followed by an intervention does not give the same result as an intervention followed by an observation. This paper contains three main contributions:

- Firstly, we show that the well-known mutilation and augmentation methods [11] for handling interventions proposed in the framework of probabilistic causal graphs have natural counterparts in OCF-based networks.
- Secondly, we propose an OCF-based counterpart of an efficient method [3] for handling sequences of interventions and observations by directly performing equivalent transformations on the causal graph.
- Finally, we highlight the interest of reasoning with sequences of observations and interventions on alert correlation, a major problem in computer security.

Let us first provide basic backgrounds on OCF networks.

2 A Brief Refresher on OCF-Based Networks

Ordinal conditional functions (OCFs) [12] is an ordinal framework for representing and changing agents' beliefs. In the following, $V = \{X, A_1, A_2, \dots, A_n\}$ denotes the set of variables. D_{A_i} denotes the domain of a variable A_i and a_i a possible instance of A_i . $\Omega = \times_{A_i \in V} D_{A_i}$ denotes the universe of discourse. An interpretation

¹ Attacks which overwhelm servers with huge number of requests.

² For instance, using a bad configuration of an application on the server machine.

$w=(a_1, a_2, \dots, a_n)$ is an instance of Ω while $w[A_i]$ denotes the value of variable A_i in w . ϕ, φ denote subsets of Ω , called events.

An OCF (also called a ranking or kappa function) denoted κ is a mapping from the universe of discourse Ω to the set of ordinals (here, we assume to a set of integers) [6]. $\kappa(w_i)$ is called a disbelief degree (or degree of surprise). By convention, $\kappa(w_i)=0$ means that w_i is not surprising and corresponds to a normal state of affairs while $\kappa(w_i)=\infty$ denotes an implausible event. The relation $\kappa(w_i)<\kappa(w_j)$ means that w_i is more plausible than w_j . The function κ is normalized if there exists at least one possible interpretation $w \in \Omega$ such that $\kappa(w)=0$. The disbelief degree $\kappa(\phi)$ of an arbitrary event $\phi \subseteq \Omega$ is defined as follows:

$$\kappa(\phi) = \min_{w_i \in \phi} (\kappa(w_i)). \quad (1)$$

Conditioning is a fundamental notion for updating a priori beliefs when a new evidence (a completely sure event) arrives. It is defined as follows (we assume that $\kappa(\phi) \neq \infty$):

$$\kappa(w_i|\phi) = \begin{cases} \kappa(w_i) - \kappa(\phi) & \text{if } w_i \in \phi; \\ \infty & \text{otherwise.} \end{cases} \quad (2)$$

The effect of conditioning is to exclude every interpretation w_i which does not satisfy the evidence ϕ while the other interpretations are decreased by $\kappa(\phi)$. In particular, the most plausible interpretation satisfying ϕ (namely, $w_j = \operatorname{argmin}_{w_i \in \phi} (\kappa(w_i))$) is assigned 0.

2.1 Causal OCF-Based Networks

Graphical models such as probabilistic networks [8] are well-known and efficient modeling and reasoning tools. Like Bayesian networks, OCF-based ones consist of two components: **i) A graphical component** consisting in a directed acyclic graph (DAG) where the nodes denote the domain variables and arcs encode direct influence relations existing between these variables, and **ii) A numerical component** composed of a set of conditional ranking functions weighting the influence endured by each variable A_i in the context of its parents U_{A_i} .

The normalization condition requires that every local ranking function should satisfy the following condition:

$$\min_{a_i \in D_{A_i}} (\kappa(a_i|u_{A_i})) = 0. \quad (3)$$

The joint ranking function encoded by a network G is computed as follows:

$$\kappa_G(a_1, a_2, \dots, a_n) = \sum_{i=1}^n \kappa(a_i|u_{A_i}). \quad (4)$$

In a *causal* OCF-based network, the graph only encodes causal (cause-effect) relationships. Hence, in a causal OCF-network, the parent set U_{A_i} of a node A_i represents all the direct causes of A_i . The following example will be used in the rest of this paper to illustrate our contributions:

Example

This example is about mechanics where we are only interested in the car startup problem. We define the following variables:

- S (for *Start*) taking its values in the domain $D_S = \{Yes, No\}$.
- B (for *Battery*) taking its values in $D_B = \{Charged, Discharged\}$.
- F (for *Fuel*) taking its values in $D_F = \{Empty, NotEmpty\}$ where the value *Empty* denotes an empty fuel tank while *NotEmpty* denotes a non empty fuel tank.
- H (for *Headlights*) taking its values in the domain $D_H = \{On, Off\}$ where the value *On* denotes that the headlights were left switched on overnight and *Off* denotes the fact that the headlights were left switched off overnight.

The OCF-based network representing the car startup problem is given in Figure 1. For instance, for the fuel variable F , the most common state is that the fuel tank is not empty while the state *Empty* is exceptional. Similarly, *Off* is the most common state for the headlights variable H . Regarding the variable B , if the headlight were left switched on overnight, then the value *Discharged* is the most common state for variable B . Lastly, if the battery is discharged or the fuel tank is empty, then the most plausible state for the start variable S is *No* (the car does not start).

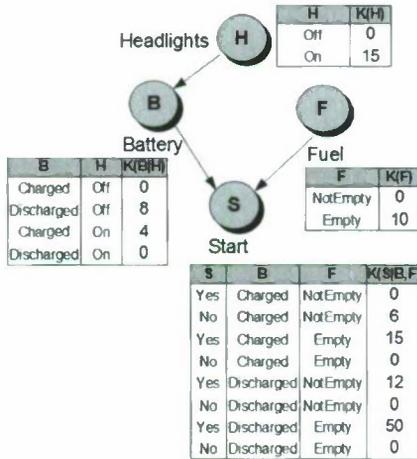


Fig. 1. The OCF-based network of the car startup problem

3 Handling Interventions in OCF Causal Networks

Interventions [11] constitute a fundamental notion for causality ascription as they provide a natural way for understanding causation. Indeed, causal relationships are more easily identified if one can directly intervene on the system (as an

experimenter) and evaluate the effects of such manipulations. An intervention is the action of forcing a variable to a specific value. It is important to note that an intervention is due to something outside the considered system and it does not matter how the intervention happens. In the example of Figure 1, we can for instance remove the spark plugs in order to prevent the car engine from starting even if the battery is charged and the fuel tank is not empty. In causal networks, an intervention on a variable A_i must not change our beliefs (expressed in some uncertainty framework) on parents U_{A_i} of A_i . There are mainly two equivalent methods for handling interventions in causal graphical models: *graph mutilation* proposed by Pearl and Verma in [13] and *graph augmentation* proposed in [10] by Pearl. In [2], the authors proposed possibilistic counterparts for the mutilation and augmentation methods. In the following, we propose the counterparts of these methods for OCF-based networks.

3.1 Handling Interventions by Mutilating the OCF Causal Network

Let G be an initial OCF-based network. An intervention on a variable A_i denoted $do(a_i)$ ensures that our beliefs on U_{A_i} (the set of parents of A_i) are not affected. In the mutilation method, this is achieved by removing all the arcs from each variable composing U_{A_i} to A_i while maintaining the rest of the graph unmodified. The obtained graph is called the *mutilated graph* and denoted G_m such that $\kappa_G(w|do(a_i)) = \kappa_{G_m}(w|a_i)$, where κ_G (resp. κ_{G_m}) is the joint ranking function encoded by G (resp. G_m). In order to determine the effect of the intervention $do(a_i)$ on the rest of the initial graph G , one can apply conditioning on the mutilated graph G_m after having observed the event $A_i = a_i$. This result is formalized in the following proposition:

Proposition 1. Let G be an OCF-based causal network and κ_G the joint ranking function encoded by G . Let G_m be the mutilated graph obtained after handling an intervention $do(a_i)$ and κ_{G_m} the joint ranking function encoded by G_m . Let also $\kappa_{G_{a_i}}$ denote the joint ranking function obtained by conditioning κ_G with $do(a_i)$. Then $\forall w \in \Omega$, $\kappa_G(w|do(a_i)) = \kappa_{G_{a_i}}(w) = \kappa_{G_m}(w|a_i)$.

3.2 Handling Interventions by Augmenting the OCF Causal Network

The principle of the augmentation method [10] is to consider an intervention as an extra node in the system. Then a parent node denoted DO_i is added to the node A_i under intervention. Hence, the parents set of the variable A_i (i.e. U_{A_i}) is augmented by the extra node DO_i allowing to specify the behavior of the variable A_i . The domain of DO_i is $\{\{do_{a_i} : \forall a_i \in D_{A_i}\}, do_{i-noact}\}$ where the value $do_{i-noact}$ means that no intervention is performed on A_i while do_{a_i} means that the variable A_i is forced to take the value a_i . The obtained augmented network is denoted G_a such that $\kappa_G(w|do(a_i)) = \kappa_{G_a}(w|DO_i = do_{a_i})$, where κ_G (resp. κ_{G_a}) is the joint ranking function encoded by G (resp. G_a).

Proposition 2. Let G be an OCF-based causal network and κ_G the joint ranking function encoded by G . Let G_a be the augmented graph for handling an intervention $do(a_i)$ by adding the node DO_i . Let $U'_{A_i} = U_{A_i} \cup DO_i$ and u'_{A_i} be an instance of $D_{U'_{A_i}}$. G_a is such that every variable A_j different from A_i has the same local ranking function as in G and

$$\kappa(a_i|u'_{A_i}) = \begin{cases} 0 & \text{if } DO_i = do_{a_i} \\ \kappa(a_i|u_{A_i}) & \text{if } DO_i = do_{i-noact} \\ \infty & \text{otherwise} \end{cases} \quad (5)$$

Then $\forall w \in \Omega$, $\kappa_G(w|do(a_i)) = \kappa_{G_a}(w|DO_i = do_{a_i})$.

4 Handling Sequences of Interventions/Observations

Contrary to the handling of a sequence involving only observations or only interventions, handling sequences involving both observations and interventions should be done differently depending on the order in which observations and interventions occur. More particularly, given an OCF-based network encoding the initial beliefs, there might exist situations where the revised beliefs after having an observation followed by an intervention will not be the same as if we have first the intervention preceding the observation. In order to illustrate this issue, consider the following two scenarios on the example of Figure 1:

Example (Continued)

1. **Scenario 1 (An observation preceding an intervention):** Assume that one morning, the car does not start. We change our a priori beliefs (the battery is working (charged), the fuel tank is not empty and the car headlights were not left switched on overnight). According to the beliefs encoded by the network of Figure 1, we deduce that either the battery is discharged or the fuel tank is empty. After this observation, assume an intervention preventing the car from starting (for example, removing a spark plug). Clearly, after this intervention, we will not change our beliefs regarding the battery and the fuel tank.
2. **Scenario 2 (An intervention preceding an observation):** Assume in this scenario that before trying to start the car, we first remove a spark plug. Unsurprisingly, the car does not start. Knowing that a plug spark was removed, it is clear that the fact that the car does not start is due to the intervention. Consequently, the most plausible state (according to Figure 1) is that the battery is *Charged* and the fuel tank is *NotEmpty* and the headlights were left switched *Off* overnight, namely the initial beliefs before any intervention or observation.

Clearly, these scenarios show that the order of occurrence of observations and interventions should be taken into account. However, existing approaches [11][2] confuse the notions of observations and interventions and do not explicitly distinguish between the two scenarios. The following section presents the OCF-based

counterpart of an efficient method for handling sequences of observations and interventions proposed in [3] (resp. in [1]) in the context of min-based (resp. product-based) causal possibilistic networks.

4.1 Graphical Handling of Sequences of Both Interventions and Observations in Causal OCF-Based Networks

Our method views each observation $A_i=a_i$ or intervention $do(a_i)$ as a belief change process that transforms an initial ranking function κ (associated with some OCF-based network) into a new distribution $\kappa(\cdot|A_i=a_i)$ or $\kappa(\cdot|do(a_i))$. Hence, it is enough to build an OCF-based network associated with $\kappa(\cdot|A_i=a_i)$ and $\kappa(\cdot|do(a_i))$. While the handling of interventions is straightforward in causal networks by mutilating the graph (or equivalently by augmenting the graph), handling graphically observations needs more operations. In the following we propose a graphical counterpart for the conditioning operation for handling observations in causal OCF-based networks. We restrict ourself to OCF-based networks where DAG's are trees, where a node can have at most one parent.

4.2 Graphical Counterpart of Conditioning for Handling Observations

In order to perform conditioning directly on the graph, conditioning is viewed as a sequence of two operations: i) A combination operation (which combines the original ranking function with the one associated with the observation $A_i=a_i$), and ii) A normalization operation (for normalizing the ranking function obtained after the combination step in case where this latter becomes sub-normalized). To make this decomposition clear, let G be an OCF-based network and κ_G be the ranking function encoded by G (κ_G is obtained from G using the chain rule of Equation 4). In order to perform the combination operation, let us define the local ranking function associated with the observation as follows:

$$\forall w \in \Omega, \kappa_{A_i=a_i}(w) = \begin{cases} 0 & \text{if } w[A_i] = a_i \\ \infty & \text{otherwise} \end{cases} \quad (6)$$

Combining the initial ranking function κ_G with $\kappa_{A_i=a_i}$ can be defined as follows:

$$\forall w \in \Omega, \kappa_{G2}(w) = \kappa_G(w) + \kappa_{A_i=a_i}(w). \quad (7)$$

The ranking function κ_{G2} is obtained from κ_G by considering as completely impossible every interpretation w where the value of A_i is different from a_i (namely, $\forall w \in \Omega \kappa_{G2}(w) = \infty$ if $w[A_i] \neq a_i$), and preserving unchanged the disbelief degrees of all interpretations w where the value of A_i is a_i . After this step, κ_{G2} may be sub-normalized. Let us define the normalization operation as follows:

$$\forall w \in \Omega, \kappa_{G3}(w) = \kappa_{G2}(w) - \min_{w \in \Omega} \kappa_{G2}(w). \quad (8)$$

Hence, using the combination and normalization formulas (see Equations 7 and 8), the conditioning given by Equation (2) can be redefined as follows:

$$\forall w \in \Omega, \kappa_G(w|A_i = a_i) = \kappa_{G3}(w). \quad (9)$$

Let us now provide the graphical counterparts of combination and normalization operations.

Graphical Counterpart of the Combination Operation. Let us use $G2$ to denote the result of integrating the new observation $A_i=a_i$ in the network G , namely the network associated with the ranking function given by Equation 7. $G2$ is specified as follows:

Proposition 3. The OCF-based network $G2$ associated with the ranking function given by Equation 7 is obtained from network G as follows:

- the structure of $G2$ is obtained from the DAG of G by deleting the arc from the parents of A_i to A_i .
- the local function of any variable A_j in $G2$ different from A_i and U_{A_i} is identical to A_j 's local function in G . Regarding, A_i and its parent denoted D , the new local ranking functions are defined as follows:
 - $\forall a_i \in D_{A_i}$,

$$\kappa_{G2}(a_i) = \begin{cases} 0 & \text{if } A_i = a_i \\ \infty & \text{otherwise} \end{cases}$$

- Let C be the parent of D in G , then $\forall d_l \in D_D, \forall c_j \in D_C \kappa_{G2}(d_l|c_j) = \kappa_G(d_l|c_j) + \kappa_G(a_i|d_l)$

The new local ranking function relative to the variable A_i ensures that only the instance a_i is fully accepted and all the other instances are completely implausible. Note that contrary to handling interventions, the ranking function relative to variable D (parent of A_i) is altered in order to ensure that disbelief degrees of every interpretation w satisfying a_i are identical in κ_G and κ_{G2} . Hence, since the value of the variable of A_i is now fully determined, there is no need to maintain the arc from the parent of A_i (here D) to A_i . One can easily check that $\forall w \in \Omega, \kappa_{G2}(w) = \kappa_G(w) + \kappa_{A_i=a_i}(w)$.

Example (Continued)

We continue with the example of Figure 1 but restricted to a tree by discarding node B (*Battery* variable) and H (*Headlights* variable). Figure 2 gives the initial network G and $G2$ obtained after combining G with the observation $S=No$. As for node F of network $G2$ of Figure 2, the new ranking function of the parent of the observed variable may be sub-normalized, the following step deals with this problem.

Graphical Counterpart of the Normalization Operation. After the combination step, the ranking function relative to the parent variable (denoted here D) of the observed one (here A_i) may be sub-normalized. Namely, it may exist an instance c_j of the parent variable of D denoted C such that $\min_{d_j \in D_D} (\kappa_{G2}(d_l|c_j)) = \beta$ ($\beta > 0$). We want to compute a new OCF network, denoted $G3$, such that it satisfies Equation 8. The network $G3$ is constructed such

that all of its local ranking functions are normalized. $G3$ is obtained by progressively normalizing local ranking functions for each variable. We first study the case where only the local ranking function on the root variable in $G2$ is sub-normalized:

Proposition 4. Let $G2$ be the network obtained from the combination step.

Assume that only the root variable, denoted by D , is sub-normalized. Let $\min_{d_l \in D_D}(\kappa_{G2}(d_l)) = \beta$ and $0 < \beta$. $G3$ is such that:

- The structure of $G3$ is identical to the one of $G2$,
- $\forall X, X \neq D, \forall x \in D_X, \forall u_x \in D_{U_X}, \kappa_{G3}(x|u_x) = \kappa_{G2}(x|u_x)$,
- $\forall d_l \in D_D, \kappa_{G3}(d_l) = \kappa_{G2}(d_l) - \beta$.

Then, $\forall \omega \in \Omega, \kappa_{G3}(\omega) = \kappa_{G2}(\omega) - \min_i(\kappa_{G2}(\omega_i))$.

After this transformation, the local ranking function relative to D is re-normalized while the joint one encoded by the network $G3$ satisfies Equation 8.

Example (Continued)

Figure 2 shows that the local ranking function relative to the root node F of network $G2$ (obtained after the combination of network G with the observation $S = No$) is sub-normalized. The normalization of this ranking function according to Proposition 4 gives the network $G3$ of Figure 2. One can easily check that the joint ranking function encoded by network $G3$ satisfies Equation 8.

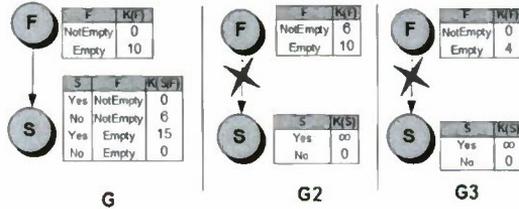


Fig. 2. Initial OCF-based network G and $G2$ (resp. $G3$) obtained after the combination (resp. normalization) step

Let us now deal with the case where the sub-normalized function is relative to a variable D which is not a root. Let us denote by C the parent of D . In this case, the ranking function of C must be altered in order to keep unchanged the underlying joint function. The normalization of a non root variable D is performed using Proposition 5 without changing the global ranking function:

Proposition 5. Let $G2$ be the network obtained from the combination step. Let D denote the variable whose ranking function is sub-normalized. Let C be the parent variable of D and c_k be the value of C such that $\min_{d_l \in D_D}(\kappa_{G2}(d_l|c_k)) = \beta$ with $0 < \beta$. Network $G3$ is such that it has the same structure as $G2$ and,

- $\forall X, X \neq D$ and $X \neq C, \kappa_{G3}(x|u_X) = \kappa_{G2}(x|u_X)$,

$$- \forall d_i \in D_D, \forall c_j \in D_C,$$

$$\kappa_{G3}(d_i|c_j) = \begin{cases} \kappa_{G2}(d_i|c_j) - \beta & \text{if } c_j = c_k \\ \kappa_{G2}(d_i|c_j) & \text{otherwise} \end{cases}$$

$$- \forall c_j \in D_C, \forall u_{c_j} \in D_{U_C},$$

$$\kappa_{G3}(c_j|u_{c_j}) = \begin{cases} \kappa_{G2}(c_j|u_{c_j}) + \beta & \text{if } c_j = c_k \\ \kappa_{G2}(c_j|u_{c_j}) & \text{otherwise} \end{cases}$$

Then, $\forall w \in \Omega, \kappa_{G2}(w) = \kappa_{G3}(w)$.

As it is shown on the example of Figure 3 (see variable *B* of network *G2*), if after the re-normalization of *D*, its parent *C* become in turn sub-normalized, then the normalization process should be repeated until reaching a root variable. Once a root is reached, it is enough to re-normalize according to Proposition 4 to get an OCF-based network where all the local ranking functions are normalized.

Example (Continued)

Here, the network *G* is limited to variables *S*, *B* and *H*. Figure 3 shows that the local ranking function relative to the non root node *B* of network *G2* (obtained after the combination of network *G* with the observation *S*=*No*) is sub-normalized. The normalization of this ranking function according to Proposition 5 gives the network *G3-a* of Figure 3. Now the normalization of *B* renders *H* sub-normalized. This latter is normalized according to Proposition 4 giving the network *G3-b* of Figure 3. 2. One can check that the joint ranking function encoded by network *G3-b* satisfies Equation 8 and *G3-b* is completely normalized. We provide in the following an application scenario of OCF-based causal networks in the area of computer security.

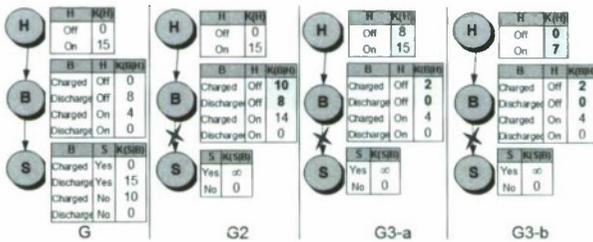


Fig. 3. Initial OCF-based network *G* and *G2* (resp. *G3-a* and *G3-b*) obtained after the combination (resp. normalization) step

5 Application to Predicting/Preventing Dangerous Attacks

Alert correlation [5] plays an important role in nowadays computer security infrastructures. It consists in analyzing the alerts triggered by one or multiple intrusion detection systems and security tools in order to provide a *synthetic* and *high-level* view of the *interesting* malicious events targeting the information system. In this application, we are concerned with *predicting/preventing* severe attacks which often are the final step in multi-step attacks. Clearly, there is a need for i) an easy *elicitation method* in order to allow security administrators to express their domain knowledge (on the security threats, vulnerabilities, etc.) and ii) a *method* to reason given observations (data directly collected from the information systems) and interventions (after manipulations and actions undertaken by the administrators, attackers, etc.). OCF-based causal networks offer several advantages for the severe attack prediction/prevention problem since it makes it easy for the administrators to elicit their knowledge and allows them to assess the plausibility that an event occur, that an attacker reaches a given objective given some observed events, etc. It also allows them to determine which countermeasures should be taken in order to prevent a given attack.

An OCF-Based Model for Severe Attack Prediction/Prevention. We are interested in anticipating severe attacks in order to prevent them by taking the appropriate countermeasures (such as preventing the suspected user from following his attack). The actions that may be undertaken by attackers and their possible consequences, the security policy and the countermeasures taken by security administrators, etc. clearly involve causal relationships that can be modeled by a causal network which can be used for instance to evaluate the plausibility of different scenarios. We propose a model for this problem and we define the following variable categories

- *Observational/interventional variables*: They represent relevant variables for monitoring the information system. For instance, the number of *HTTP* requests sent to a server represents a relevant information for detecting/preventing denial of service attacks.
- *Attack objective variables*: They represent the final/intermediate objectives targeted by the attackers. For example, *gaining a local access, a root access*, etc. are among the most recurrent objectives of nowadays internet hackers.

In this model, *observational/interventional variables* are either directly observed or manipulated (for instance, a network monitor can count the number of inbound *HTTP* requests, some variables can however be manipulated by the administrators' interventions such as configuring a firewall to stop the requests coming from a given suspected host...) while *attack objective variables* are associated with the attacks administrators may want to predict/prevent. While the network structure easily encodes the causal relationships between the relevant variables, the a priori and conditional ranking functions allow to easily weight the uncertainty and the influence of each variable on its children.

5.1 Scenario Evaluation and Countermeasure Determination

After an OCF-based network is built based on the domain knowledge, it can then be efficiently used for different tasks. In particular, it can be used for

i) Scenario evaluation: Given an OCF-causal network representing the administrators' knowledge, one can evaluate the plausibility of any event of interest such as the one that an attacker reaches a given attack objective having observed some security events in the audit data.

ii) Countermeasure determination: The aim of this task is to determine what action(s) should be taken in order to prevent an attacker from attaining a given objective. Administrators can intervene on some variables and assess the plausibility that a given attack objective is attained in order to determine whether this action is adequate or insufficient for preventing from this attack.

It is obvious that there is a need in this application before actually taking countermeasures to intervene on the model (instead of directly intervening on the system) in order to check whether a given intervention (here a countermeasure) will aid to secure the information system or allow an attacker to attain his objective, etc. By evaluating different scenarios, the users can determine the most appropriate countermeasures. Finally, note that it is important to take into account the order of arrival of observations/interventions. For instance, for a security administrator, observing a Web server crash then intervening on the system by stopping the network will need lead to the same conclusions as first stopping the network then observing the Web server crash. Clearly, our approach for handling sequences of both observations and interventions is relevant for the severe attack prediction/prevention problem.

6 Conclusion

This paper addressed important issues regarding belief change in OCF-based networks and handling sequences of both observations and interventions. It provided three major contributions: a) We showed that the well-known graph mutilation and augmentations methods for handling interventions in probabilistic graphs have natural counterpart in OCF networks. b) We proposed an OCF-based counterpart of an efficient method for handling observations in causal graphs by directly performing equivalent transformations on the initial graph. This method allows to efficiently integrate new observations and providing a graphical counterpart for the conditioning operation. c) We provided a real application scenario in the field of computer security highlighting the importance of reasoning in presence of sequences of observations and interventions.

Acknowledgements

This work is supported by the PLACID project (<http://placid.insa-rouen.fr/>).

References

1. Benferhat, S.: Interventions and belief change in possibilistic graphical models. *Artificial Intelligence* 174(2), 177–189 (2010)
2. Benferhat, S., Smaoui, S.: Possibilistic causal networks for handling interventions: A new propagation algorithm. In: *The 22nd AAAI Conference on Artificial Intelligence*, pp. 373–378 (2007)
3. Benferhat, S., Tabia, K.: Min-based causal possibilistic networks: Handling interventions and analyzing the possibilistic counterpart of Jeffrey’s rule of conditioning. In: *Proceedings 19th European Conference on Artificial Intelligence - ECAI 2010, Lisbon Portugal*, p. 6 (August 2010)
4. Darwiche, A., Pearl, J.: On the logic of iterated belief revision. *Artif. Intel.* 89, 1–29 (1996)
5. Debar, H., Wespi, A.: Aggregation and correlation of intrusion-detection alerts. In: Lee, W., Mé, L., Wespi, A. (eds.) *RAID 2001. LNCS*, vol. 2212, pp. 85–103. Springer, Heidelberg (2001)
6. Goldszmidt, M., Pearl, J.: Rank-based systems: A simple approach to belief revision, belief update, and reasoning about evidence and actions. In: *3rd Intern. Conf. on Principles of Knowledge Representation and Reasoning*, Cambridge, MA, pp. 661–672 (1992)
7. Halpern, J.Y.: Conditional plausibility measures and bayesian networks. *J. Artif. Int. Res.* 14(1), 359–389 (2001)
8. Jensen, F.V., Nielsen, T.D.: *Bayesian Networks and Decision Graphs*. Information Science and Statistics. Springer, Heidelberg (June 2007)
9. Ma, J., Liu, W.: A general model for epistemic state revision using plausibility measures. In: *2008 Conference on ECAI*, pp. 356–360 (2008)
10. Pearl, J.: (bayesian analysis in expert systems): Comment: Graphical models, causality and intervention. *Statistical Science* 8(3), 266–269 (1993)
11. Pearl, J.: *Causality: Models, Reasoning, and Inference*. Cambridge University Press, Cambridge (March 2000)
12. Spohn, W.: Ordinal conditional functions: A dynamic theory of epistemic states. In: *Causation in decision, belief change, and statistics*, vol. II, pp. 105–134. Kluwer Academic Publishers, Dordrecht (1988)
13. Verma, T., Pearl, J.: Equivalence and synthesis of causal models. In: *UAI 1990: Proc. of 6th Annual Conf. on Uncertainty in Artificial Intelligence*, pp. 255–270. Elsevier Science Inc., Amsterdam (1991)

A Context-Sensitive Manifold Ranking Approach to Query-Focused Multi-document Summarization

Xiaoyan Cai and Wenjie Li

Department of Computing, The Hong Kong Polytechnic University
{csxcai, cswjli}@comp.polyu.edu.hk

Abstract. Query-focused multi-document summarization aims to create a compressed summary biased to a given query. This paper presents a context-sensitive approach based on manifold ranking of sentences to this summarization task. The proposed context enhanced manifold ranking approach not only looks at the sentence itself, but also considers its surrounding contextual information. Compared to the existing manifold ranking approach which totally ignores the contextual information of a sentence, this approach can capture more additional relevant information which is especially necessary for formulating the relationships between short text snippets like sentences. Experiments are conducted on the DUC 2005 and DUC 2006 data sets and the ROUGE evaluation results demonstrate the advantages of the proposed approach.

Keywords: Query-focused multi-document summarization, context-sensitive manifold ranking.

1 Introduction

With the growing popularity of the Internet and a variety of information services, obtaining the desired information has become a serious problem in the information age. As such, new technologies that can process information efficiently are needed. Automatic document summarization, which is the process of reducing the size of documents while preserving the important semantic content, is an essential technology to overcome this obstacle. Most of the summarization work done till date follow the sentence extraction framework, which ranks sentences in some way and selects top-ranked sentences from original documents to form summaries. Extractive summarization generally falls into two categories according to the nature of summarization. They are generic summarization, which aims at extracting a summary about general ideas of documents and query-focused summarization, which aims at not only extracting the important information contained in the documents, but also guaranteeing that the extracted information is biased to the given query. What we are interested in this paper is query-focused summarization.

Query-focused multi-document summarization has drawn much attention in recent years due to its applicability and merits in real-world applications. Since it is able to provide concise information corresponding to the specific queries from the different users, it has been applied to the services like personalized Web service or document

understanding to support the various interests of users. In contrast to the conventional task of question answering (QA) that mainly focuses on simple factoid questions and results in precise answers such as person, location or date, etc., in the case of query-focused summarization, the queries are mostly real-world complex questions (e.g., “Identify and describe types of organized crime that crosses borders or involves more than one country.”). Such complex questions make summarization tasks more challenge and meanwhile have a very important role to play.

Recently, manifold ranking algorithm has been exploited for query-focused multi-document summarization, such as in [1]. The manifold ranking based approaches first constructed a weighted graph representing query and sentences as vertices. Then the positive ranking score of query was iteratively propagated to nearby vertices via the structure of the graph. Finally all sentences were ranked according to their ranking scores, with a larger score indicating higher relevance. Inspired by the success of manifold ranking, in this paper we propose an enhanced approach to further integrate the contextual information of sentences into manifold ranking for query-focused multi-document summarization. The motivation to this approach is the consensus that short text snippets like sentences often contain insufficient information to measure the relationships between them and to support ranking of them. In our approach, we use one preceding and one following sentences of the sentence currently under concern as the additional contextual information to enrich the sentence representation or to refine the standard sentence-to-sentence cosine similarity measure and develop four strategies to construct the context-sensitive affinity matrixes, which are essential to a manifold ranking algorithm. Compared to the existing manifold ranking approach, our approach can capture more additional relevant information by using contextual sentences. The experiments conducted on the data sets from DUC 2005 and DUC 2006 show that the summarization results with contextual information are better than that those without contextual information, achieving the state-of-the-art performance.

The remainder of this paper is organized as follows. Section 2 reviews related work. Section 3 introduces the proposed manifold ranking algorithm using contextual information of sentences. Section 4 then presents experiments and evaluations. Finally, Section 5 concludes the paper.

2 Related Work

A variety of summarization approaches have been proposed in the literature. These approaches were either extractive or abstractive. Extractive summarization assigned a significance score to each sentence and extracted the sentences with highest scores to form the summaries. Abstraction summarization, on the other hand, involved a certain degree of understanding of the content conveyed in the original documents and created the summaries based on information fusion and/or language generation techniques [2]. Like most researchers in this field, we follow the extractive summarization framework in this work.

Under the framework of extractive summarization, sentence ranking is the issue of most concern. Traditional feature-based approaches evaluated sentence significance and ranked the sentences depending on the features that were well-designed to characterize the different aspects of the sentences. The centroid-based approach [3] was one

of the most popular feature-based summarization approaches. Other statistical and linguistic features such as term frequency, sentence position, sentence dependency structure and query relevance etc. have also been extensively investigated in the past. In recent years, graph-based approaches have been proposed to rank sentences. These approaches modeled a document or a set of documents as a weighted text graph, took into account the global information and recursively calculated sentence significance from the entire text graph rather than only relying on the unconnected individual sentences. LexRank [4] and TextRank [5] were examples of such approaches. Both of them were motivated by PageRank, which has been successfully used for ranking Web pages in the Web Graph.

Most existing query-focused multi-document summarization approaches incorporated the information of the given query into the generic summarizers in order to extract the sentences suiting the user's declared information need. In [6], a query-based feature that computed the similarity between sentence and query was combined with a set of document-based features. The role of the query words and the named entities appeared in the query are especially emphasized in [7]. Later, a topic-sensitive version of PageRank was proposed to incorporate the relevance of a sentence to the query into LexRank to get a biased PageRank ranking [8]. As a matter of fact, for those graph-based approaches, the influence of the query was normally reflected in the formulation of sentence vertices in a text graph.

Different from the traditional query-focused summarization approaches, which were usually the simple extensions of generic summarizers and did not uniformly fuse the information in the query and the documents, Wan et al. [1] proposed a manifold ranking based approach to make uniform use of sentence-to-sentence and sentence-to-query relationships. A weighted graph was built where the vertices included both the query description and the sentences in the documents. The manifold ranking was employed to iteratively propagate the relevance of the query to nearby vertices via the graph structure. The ranking score of a sentence obtained by this process indicated the topic-biased informativeness of the sentence and those with high ranks are chosen to form the summary.

3 Context-Sensitive Manifold Ranking Approach

Manifold ranking is a semi-supervised learning that explores the relationship among all the data points in the feature space [9, 10]. It has two versions regarding the different tasks: (1) to rank the data points, or (2) to predict the labels of the unlabeled data points. For the task of ranking, the prior assumptions of it include (1) nearby points are likely to have the same ranking scores; and (2) points on the same structure (typically referred to as a cluster or a manifold) are likely to have the same ranking scores.

3.1 Notation

In this paper, each sentence, either a document sentence or a query sentence, is represented by an m dimensional feature vector x and forms a sentence point in the Euclidean space. Let $\mathcal{X} = \{x_0, x_1, \dots, x_n\} \subset R^m$, where the first point x_0 is the query description and the rest n points are the sentences to be ranked according to their relevance to the

query. Note that because the topic description is usually short, in our experiments we treat it as a pseudo-sentence and it is processed in the same way as the other sentences. Let $y = [y_0, \dots, y_n]^T$, where $y_0 = 1$ corresponding to the query sentence x_0 and $y_i = 0, (i = 1, \dots, n)$ for all the sentences in the documents. Let $f : \mathcal{X} \rightarrow \mathcal{R}$ denote a ranking function which assigns to each sentence point x_i a ranking score f_i .

3.2 Basic Manifold Ranking Algorithm

The basic manifold ranking algorithm is presented in Table 1. An intuitive description of this algorithm is: a weighted graph is first formed which takes each sentence as a vertex; assign a positive ranking score, to the query while zero to the remaining sentences; all the vertices then spread their scores to the nearby vertices via the weighted graph; the spread process is repeated until a global stable state is reached, and all the vertices except the query will have their own scores according to which they will be ranked. The propagation of ranking score reflects the relationship of all vertices, since in the weighted graph, distant vertices will have different ranking scores unless they belong to the same cluster consisting of many points that help to link the distant points, and nearby vertices will have similar ranking scores unless they belong to different clusters. In the context of our task, there is only one query in the query set. The resultant ranking score of a sentence in the document is in proportion to the probability that it is relevant to the query, with large ranking score indicating high probability.

Table 1. Basic Manifold Ranking Algorithm

1. Sort the cosine similarities among vertices in ascending order. Repeat connecting the two vertices with an edge according to the order until a connected graph is obtained.
2. Form the affinity matrix W by cosine similarities measure between any two vertices, if there is an edge linking the two vertices. Let $W_{ii} = 0$.
3. Symmetrically normalize W by $S = D^{-\frac{1}{2}} W D^{-\frac{1}{2}}$ in which D is the diagonal matrix with (i, i) -element equal to the sum of the i th row of W .
4. Iterate $f(t+1) = \alpha S f(t) + (1-\alpha)y$ until convergence, where α is a parameter in $[0,1)$, and y is the original labeling.
5. Let f^* denotes the limits of the sequence $\{f(t)\}$. Rank each sentence according to its ranking score in f^* .

In the above iterative algorithm, the normalization in the third step is necessary to prove the algorithm's convergence. During the fourth step, each sentence point receives the information from its neighbors (first term), and also retains its initial information (second term). The parameter of manifold ranking weight α specifies the

relative contributions to the ranking scores from neighbors and the initial ranking scores. Self-reinforcement is avoided, therefore the diagonal elements of the affinity matrix are set to zero.

The theorem in [10] guarantees that the sequence $\{f(t)\}$ converges to

$$f^* = \beta(I - \alpha S)^{-1} y \quad (1)$$

where $\beta = 1 - \alpha$.

3.3 Context-Sensitive Affinity Matrix

A key part in the above manifold ranking algorithm is the affinity matrix W . The definition of W mainly involves two essential aspects: (1) pairwise similarity metric, (2) sentence vertex representation.

In previous use, manifold ranking algorithm proposed in text processing only makes use of content words of the current sentences under concern. This sentence representation can express very limited information of each sentence and the cosine similarity calculated based on such representation may not truly reflect the similarity between the sentences. Table 2 shows a subset of a cluster in DUC 2005, and the corresponding cosine similarity matrix is shown in Table 3.

From the cosine similarity values shown in Table 3, we can see that the sentence 2 is similar to the sentence 1. However, from semantic perspective of the original document, we think the sentence 2 is much more similar to the sentence 4 than other sentences. The reason of this problem may be imputable to the fact that we ignore the contextual information of the sentences.

Table 2. The First 6 Sentences in a Subset of Cluster d311i from DUC 2005

SenNo	Text
1	International Company News: VW fails to convince GM over car factory 'copying'
2	VOLKSWAGEN has failed to convince General Motors that its plans for a revolutionary car plant in Spain are not a copy of a project drafted previously by the US group.
3	'We have a right to be sceptical,' Mr David Herman, chairman of GM's German subsidiary Adam Opel, said yesterday.
4	'It would be a real tour de force' if Mr Jose Ignacio Lopez de Arriortua, GM's former procurement chief who is now at VW, had managed to develop a new concept between mid-March, when he left the US, and mid-June when he announced VW's plans.
5	Mr Herman was responding to claims in a letter received from VW in which Mr Ferdinand Piech, chairman, said the German company did not have any confidential plans or documents about GM's ultra-low-cost factory project.
6	Mr Herman confirmed that he had written to Mr Piech before Mr Lopez's announcement, suggesting that he consider the possible consequences if VW's project were the same as the one developed at GM under Mr Lopez's direction.

Table 3. Cosine Similarities of Sentences in Table 2

SenNo	1	2	3	4	5	6
1	0	0.3081	0.0499	0.0972	0.1499	0.0745
2	0.3081	0	0.0000	0.0292	0.0749	0.0346
3	0.0499	0.0000	0	0.0533	0.2681	0.2078
4	0.0972	0.0292	0.0533	0	0.1300	0.2601
5	0.1499	0.0749	0.2681	0.1300	0	0.3515
6	0.0745	0.0346	0.2078	0.2600	0.3515	0

In order to improve the performance of summarization, we combine the contextual information into the basic manifold ranking. For this purpose, a sentence point $x_i \in R^m$ is re-defined in both the original domain using its original feature vector $x_i^s \in R^{m_s}$, and in the contextual domain by introducing $x_i^c \in R^{m_c}$, which yields m_c dimensional contextual feature vectors representing the surrounding contextual sentences. We combine one preceding and one following sentences of the current sentence as a new pseudo sentence, and deem this new pseudo sentence as the contextual information of the current sentence. Then the contextual information and the original information of the current sentence lead to two different similarity measures, which can be easily computed and combined. For example, we can sum the original and the contextual dedicated affinity matrices (e.g., W_s and W_c), or introduce the cross-information between the original and the contextual features (e.g., W_{sc} and W_{cs}) in the formulation.

In the following, we present four different strategies for joint consideration of the original and the contextual information of sentences in a unified framework for affinity matrix construction.

● *The Stacked Affinity Matrix*

The most commonly adopted strategy in affinity matrix construction for the manifold ranking algorithm is to exploit the information of an original sentence $x_i \equiv x_i^s$. However, performance can be improved by including both the original and the contextual information of the sentences. This is usually done by means of the “stacked” approach, in which the new feature vectors are built from the concatenation of the sentence and its context features.

Let us define x_i as the concatenation of the two feature vectors x_i^s and x_i^c . That is, $x_i \equiv \{x_i^s, x_i^c\}$, then the corresponding ‘stacked’ affinity matrix is:

$$W_{stacked} \equiv W(x_i, x_j) = sim(x_i, x_j) \quad (2)$$

which does not include explicit cross relations between x_i^s and x_i^c . $sim(x_i, x_j)$ is the cosine similarity between the two sentence points x_i and x_j . Table 4 below shows the cosine similarities of the sentences using stacked strategy in Table 2. From the table, we can see that this time the sentence 2 is much more similar to the sentence 4 when the additional contextual information is involved.

Table 4. Cosine Similarities of Sentences in Table 2 using Stacked Strategy

SenNo	1	2	3	4	5	6
1	0	0.2485	0.4086	0.0718	0.0719	0.0713
2	0.2485	0	0.1288	0.3153	0.1485	0.1591
3	0.4086	0.1288	0	0.1039	0.5830	0.1653
4	0.0718	0.3153	0.1039	0	0.2313	0.5178
5	0.0719	0.1485	0.5830	0.2313	0	0.3041
6	0.0714	0.1591	0.1653	0.5178	0.3041	0

- *The Direct Summation Affinity Matrix*

A simple composite affinity matrix combining the original and the contextual information can be derived from the concatenation of the original sentence affinity matrix and the contextual sentence affinity matrix. That is:

$$\begin{aligned} W_{direct}(x_i, x_j) &= W_s(x_i^s, x_j^s) + W_c(x_i^c, x_j^c) \\ &= sim(x_i^s, x_j^s) + sim(x_i^c, x_j^c) \end{aligned} \quad (3)$$

Note that $\dim(x_i^s) = m_s$, $\dim(x_i^c) = m_c$, and $\dim(W) = \dim(W_s) = \dim(W_c) = n \times n$, where \dim denotes the dimension. By this affinity matrix construction strategy, the relationships between two sentences are judged according to not only the relationship between the sentences themselves, but also the relationship between the contexts of the sentences.

- *Weighted Summation Affinity Matrix*

Alternatively the composite affinity matrix that balances the original and the contextual information in (3) can be constructed as follows:

$$\begin{aligned} W_{weighted}(x_i, x_j) &= \eta \cdot W_s(x_i^s, x_j^s) + (1 - \eta) \cdot W_c(x_i^c, x_j^c) \\ &= \eta \cdot sim(x_i^s, x_j^s) + (1 - \eta) \cdot sim(x_i^c, x_j^c) \end{aligned} \quad (4)$$

where η is a positive real-valued parameter ($0 < \eta < 1$), which constitutes a tradeoff between the original and the contextual information in forming the sentence affinity matrix. This composite affinity matrix allows us extract some information from the best tuned η parameter.

- *The Cross-information Affinity Matrix*

The preceding direct summation matrix can be conveniently modified to account for the cross relationship between the original and the contextual information. That is, it can be expressed as the sum of the four positive definite matrices, accounting for the affinity between the two sentences' original content, between their contextual sentences, and the cross-terms between the original and contextual counterparts.

$$\begin{aligned}
W(x_i, x_j) &= W_s(x_i^s, x_j^s) + W_c(x_i^c, x_j^c) \\
&+ W_{sc}(x_i^s, x_j^c) + W_{cs}(x_i^c, x_j^s) \\
&= \text{sim}(x_i^s, x_j^s) + \text{sim}(x_i^c, x_j^c) \\
&+ \text{sim}(x_i^s, x_j^c) + \text{sim}(x_i^c, x_j^s)
\end{aligned} \tag{5}$$

where $W_{sc}(x_i^s, x_j^c)$ is the cosine similarity between x_i^s and x_j^c , $W_{cs}(x_i^c, x_j^s)$ is the cosine similarity between x_i^c and x_j^s . The only restriction for this formulation to be valid is that x_i^c and x_j^s need to have the same dimension ($N_c = N_s$). This can be easily achieved as the dimension of the sentence vector is dependent on word number in document set, which is a fixed value.

Once we obtain the modified affinity matrix, we can use them to perform the manifold ranking algorithm again to improve the sentence ranking results. The overall procedure is the same as described in the ranking algorithm in Table 1.

3.4 Summary Generation and Redundancy Control

In multi-document summarization, the number of the documents to be summarized can be very large. This makes information redundancy problem appear to be more serious in multi-document summarization than in single-document summarization. Redundancy control becomes an inevitable process. Since our focus in this study is the design of effective (sentence) ranking algorithms, we apply a straightforward but effective sentence selection principle. We incrementally add into the summary the highest ranked sentence of concern if it doesn't significantly repeat the information already included in the summary until the word limitation of the summary is reached.

4 Experiments

We conduct the experiments on the data sets from the DUC 2005 and the DUC 2006. In these two years, query-focused multi-document summarization is the only task. According to the task definitions, systems are required to produce a concise summary for each document set and the length of summaries is limited to 250 English words.

A well-recognized automatic evaluation toolkit ROUGE [11] is used for evaluation. It measures summary quality by counting the overlapping units between system-generated summaries and human-written reference summaries. We report three common ROUGE scores in this paper, namely ROUGE-1, ROUGE-2 and ROUGE-SU4 which base on Uni-gram match, Bi-gram match, and unigram plus skip-bigram match with maximum skip distance of 4. Documents and queries are pre-processed by segmenting sentences and splitting words. Stop words are removed and the remaining words are stemmed using Porter stemmer.

4.1 Performance Evaluation and Comparison

In the experiments, the manifold ranking based summarizer using contextual information is compared with the two baselines employed in the DUC. They are the lead

baseline and the coverage baseline. The lead baseline takes the first sentences one by one in the last document in the collection, where documents are assumed to be ordered chronologically. The coverage baseline takes the first sentence one by one from the first document to the last document. We also present the results of top three systems with the highest ROUGE scores that participate in the DUC 2005 and the DUC 2006 for comparison.

For further comparison of the context-sensitive manifold ranking algorithm, we also implement the basic manifold ranking algorithm without using any contextual information as proposed in [1].

Tables 5 and 6 show the comparison results on the DUC 2005 and 2006 data sets respectively. The parameters of manifold ranking based approaches are set as follows: $\alpha = 0.6$. And the parameter of the weighted summation affinity matrix is set as $\eta = 0.75$. S15 and S24 etc. in the tables are the IDs of those top performing systems participated in the DUC, and the other rows show the results of the proposed approach with four different affinity matrix construction strategies and the two baselines. 'Stacked' denotes the use of stacked affinity matrix, 'Direct' denotes the use of direct summation affinity matrix, 'Weighted' denotes the use of weighted summation affinity matrix, and 'Cross' denotes the use of cross-information affinity matrix.

Table 5. Experimental Results on the Data of DUC 2005

Systems	ROUGE-1	ROUGE-2	ROUGE-SU4
Stacked	0.38592	0.07498	0.13371
Direct	0.38951	0.07501	0.13385
Weighted	0.39005	0.07515	0.13397
Cross	0.39249	0.07520	0.13405
Wan's	0.38523	0.07496	0.13353
S15	0.37665	0.07381	0.13260
S4	0.37484	0.07003	0.12798
S17	0.36930	0.07256	0.12977
Coverage Baseline	0.34659	0.06013	0.09275
Lead Baseline	0.30583	0.04875	0.08154

Table 6. Experimental Results on the Data of DUC 2006

Systems	ROUGE-1	ROUGE-2	ROUGE-SU4
Stacked	0.41702	0.10284	0.17405
Direct	0.41715	0.10291	0.17419
Weighted	0.41719	0.10295	0.17425
Cross	0.41734	0.10358	0.17430
Wan's	0.41685	0.10279	0.17401
S12	0.41611	0.10276	0.17399
S23	0.41505	0.10800	0.17834
S24	0.41020	0.10727	0.17431
Coverage Baseline	0.36753	0.08132	0.14596
Lead Baseline	0.33574	0.06942	0.12439

From Tables 5 and 6, we can see that on the two DUC data sets, the proposed approaches outperform all the top systems and the baseline systems on all the ROUGE scores. When compared with Wan’s approach, we can also see that after getting the contextual information involved in affinity matrix construction, the enhanced context-sensitive manifold ranking approach receives improved performance on both the DUC 2005 and the DUC 2006 data sets. This demonstrates the advantages using contextual information in manifold ranking.

4.2 Influence of Parameter η Used in Weighted Summation Affinity Matrix

Recall that in the definition of the weighted summation affinity matrix, the parameter η constitutes a tradeoff between the original and contextual information to form sentence affinity matrix. Figure 1 illustrates the influence of the parameter η on the summarization based on the context-sensitive manifold ranking using weighted summation affinity matrix. It is observed that when η varies from 0 to 0.7, the performances of the proposed approach are always worse than the corresponding performances of the original manifold ranking approach. It is the better case when η varies from 0.7 to 1, which demonstrates that the contextual information can help to improve the performance, but relying only on the contextual information while ignoring the original information of the sentences will unavoidably hurt the performance.

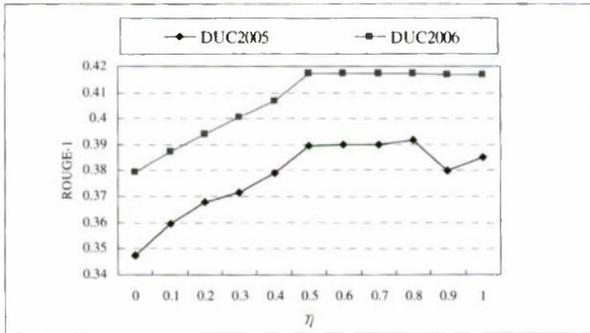


Fig. 1. ROUGE-1 vs. η

4.3 Influence of Parameter Tuning

Figure 2 and Figure 3 below demonstrate the influence of the manifold weight α in the proposed enhanced manifold ranking approach based four different affinity matrices. It is observed that the small values of α can deteriorate the summarization performance, while the performance of summarization will achieve relative stable state when α is around 0.6. It proves that the setting of α value is reasonable in the above experiments.

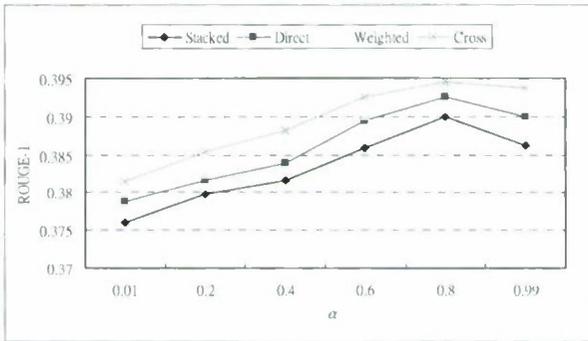


Fig. 2. ROUGE-1 vs. α in DUC 2005

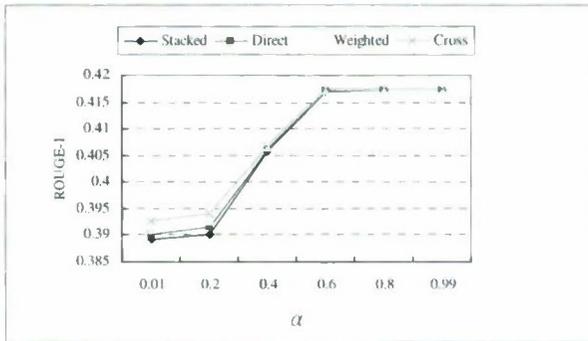


Fig. 3. ROUGE-1 vs. α in DUC 2006

5 Conclusion

In this paper, we propose a context-sensitive manifold ranking approach to multi-document summarization. Our approach takes advantage of both the original and the contextual information of the sentences from the documents. By this approach, the refined affinity matrix can capture more related information. The experimental results show that the proposed approach improves system performance and the resultant system is comparable to the top performing system in the DUC.

Acknowledgment

The work described in this paper was supported by an internal grant from the Hong Kong Polytechnic University (Account Number: G-YG80).

References

1. Wan, X.J., Yang, J.W., Xiao, J.G.: Manifold-ranking based topic-focused multi-document summarization. In: Proceedings of 20th International Joint Conference on Artificial Intelligence, pp. 2903–2908 (2007)
2. Barzilay, R., McKeown, K.R.: Sentence Fusion for Multi-document News Summarization. *Comput. Linguist.* 31(3), 297–327 (2005)
3. Radev, D.R., Jing, H.Y., Stys, M., Tam, D.: Centroid-based summarization of multiple documents. *Information Processing and Management* 40, 919–938 (2004)
4. Erkan, G., Radev, D.R.: LexRank: Graph-based centrality as salience in text summarization. *Journal of Artificial Intelligence Research* 22, 457–479 (2004)
5. Mihalcea, R.: Graph-based ranking algorithms for sentence extraction, applied to text summarization. In: Proceedings of ACL (2004)
6. Saggion, H., Bontcheva, K., Cunningham, H.: Robust generic and query-based summarization. In: EACL 2003, pp. 235–238 (2003)
7. Conroy, J.M., Schlesinger, J.D.: CLASSY query-based multi-document summarization. In: DUC 2005 (2005)
8. Otterbacher, J., Erkan, G., Radev, D.R.: Using Random Walks for Question-focused Sentence Retrieval. In: HLT/EMNLP 2005, pp. 915–922 (2005)
9. Zhou, D.Y., Bousquet, O., Lal, T.N., Weston, J., Scholkopf, B.: Learning with local and global consistency. In: Proceedings of 18th NIPS (2003)
10. Zhou, D.Y., Weston, J., Gretton, A., Bousquet, O., Scholkopf, B.: Ranking on data manifolds. In: Proceedings of 18th NIPS (2003)
11. Lin, C.Y.: ROUGE: A package for automatic evaluation of summaries. In: Proceedings of workshop on text summarization branches out, post-conference workshop of ACL 2004 (2004)

A Novel Approach to Compute Similarities and Its Application to Item Recommendation

Christian Desrosiers¹ and George Karypis²

¹ Computer Engineering & IT dept., Ecole de Technologie Superieure,
Montreal, Canada

`christian.desrosiers@etsmtl.ca`

² Computer Science & Engineering dept., University of Minnesota, Minneapolis, USA
`karypis@cs.umn.edu`

Abstract. Several key applications like recommender systems deal with data in the form of ratings made by users on items. In such applications, one of the most crucial tasks is to find users that share common interests, or items with similar characteristics. Assessing the similarity between users or items has several valuable uses, among which are the recommendation of new items, the discovery of groups of like-minded individuals, and the automated categorization of items. It has been recognized that popular methods to compute similarities, based on correlation, are not suitable for this task when the rating data is sparse. This paper presents a novel approach, based on the *SimRank* algorithm, to compute similarity values when ratings are limited. Unlike correlation-based methods, which only consider user ratings for common items, this approach uses all the available ratings, allowing it to compute meaningful similarities. To evaluate the usefulness of this approach, we test it on the problem of predicting the ratings of users for movies and jokes.

1 Introduction

Many important applications like recommender systems deal with data in the form of ratings made by users on items. In such applications, one of the most crucial tasks is to find users that share common interests, or items with similar characteristics. Assessing the similarity between users or items has several valuable uses, among which are the recommendation of new items, the discovery of groups of like-minded individuals, and the automated categorization of items.

A popular method to compute the similarity between two users, found in many collaborative filtering recommender systems, is based on the correlation between the ratings made by these users on common items. As recognized by several recent works on this topic, such as [5,18], this method is very sensitive to sparse data. For instance, while two users can be similar if they have rated different items, this method is unable to evaluate their similarity in such cases. Moreover, although recent approaches based on dimensionality reduction and graph theory have been proposed for this problem, they also have their limitations. For example, they cannot be used in situations where there are categorical

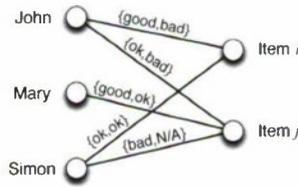


Fig. 1. A bipartite graph representing responses (sets of categorical values) given by users to items

ratings or other non-numerical rating types, such as the one shown in Figure 1, and do not provide an easy way to integrate prior information on the similarities.

This paper presents a novel approach to compute similarities between users or items when only a limited number of ratings are available. Based on the well-known algorithm *SimRank* [9], this approach models the relations between user similarities and item similarities using a system of linear equations, and computes the similarity values by solving this system. However, unlike *SimRank* and its recent extensions, our approach has the additional advantage of allowing one to evaluate the agreement between any type of ratings, and integrate prior similarity information.

The rest of this paper is organized as follows. In Section 2, we present some of the most relevant work on the topic and describe the advantages of our approach over these works. We then present the details of our approach in Section 3, and illustrate in Section 4 its usefulness on the problem of predicting the ratings of users for movies and jokes. Finally, Section 5 provides a brief summary of our work and contributions, and describes some of its possible extensions.

2 Related Work

2.1 Item Recommendation and Sparsity

Sparsity is a problem occurring frequently in recommender systems when many users have provided ratings to a limited number of items, or many items have received only a few ratings. A solution proposed for this problem consists in using item content information to enhance the computation of similarities [10,14]. However, reliable content information may not be available, for example, if obtaining this information requires expensive resources (e.g., hand made annotations) or is simply too difficult (e.g., audio or video data).

Dimensionality reduction methods have also been developed to alleviate the problem of sparsity. These methods work by decomposing the user-item rating matrix [2,17] or a sparse similarity matrix [5,6] into a limited number of latent factors. These factors, which represent high-level characteristics of users and items, are then used to predict new ratings. While decomposition approaches are among the most accurate rating prediction methods, they generally lack the ability to discover local relations in the data. Moreover, this class of techniques can only be used with numerical ratings, not categorical ones.

Another category of methods proposed for recommending items in sparse data uses graph theory to model the interactions between users and items and measure the strength of these relations. Such methods include approaches based on geodesic distance [15], diffusion kernels [11], and random walks [5,8,18]. A common problem with these methods is their lack of interpretability and the difficulty of translating ratings into link weights, for instance, if the ratings are negative or non-numerical.

Finally, a different approach, proposed in [3], computes item similarities by solving a global regression problem which finds the similarity values that best predict known ratings using an item-based nearest-neighbor formulation. This approach has three main limitations. First, it relies on a correlation-based method to compute the nearest neighbors, which may be sensitive to sparsity. Also, the item-based formulation used in this approach only considers the ratings made by common users, which also creates problems when the rating data is sparse. Finally, the item similarities computed by this method depend on the rating that is predicted, which is not suitable to the task of finding general similarities between all items.

2.2 SimRank

The method introduced in this paper is closely related to the bipartite version of the *SimRank* algorithm proposed by Jeh and Widom [9]. Let \mathcal{U} and \mathcal{I} be the two sets of nodes of a bipartite graph representing, for instance, the users and items of a recommender system. Moreover, denote by $\mathcal{I}_u \subseteq \mathcal{I}$ be the set of items purchased by a given user $u \in \mathcal{U}$, and let $\mathcal{U}_i \subseteq \mathcal{U}$ be the set of users that have purchased an item $i \in \mathcal{I}$. The similarity between two users u and v , $s(u, v)$, is obtained as the average similarity of the items purchased by these users:

$$s(u, v) = \frac{C_1}{|\mathcal{I}_u||\mathcal{I}_v|} \sum_{i \in \mathcal{I}_u} \sum_{j \in \mathcal{I}_v} s(i, j), \quad (1)$$

where $C_1 \in [0, 1]$ is a constant controlling the flow of similarity values on the graph links. Likewise, the similarity between two items i and j , $s(i, j)$, can be computed as the average similarity of users that have purchased these items:

$$s(i, j) = \frac{C_2}{|\mathcal{U}_i||\mathcal{U}_j|} \sum_{u \in \mathcal{U}_i} \sum_{v \in \mathcal{U}_j} s(u, v), \quad (2)$$

C_2 having the same role as C_1 . *SimRank* computes the similarity values by updating them iteratively using equations (1) and (2), until a fixed-point is reached.

A significant limitation of this approach, in the context of item recommendation, is that it only considers the interactions between users and items (e.g., purchases) but not the ratings. Another method called *SimRank++*, recently proposed in [1], extends *SimRank* by taking into account the link weights as modified transition probabilities. In this method, the similarity between two

nodes is computed as a weighted average of the similarities of their adjacent nodes:

$$s(u, v) = C_1 \sum_{i \in \mathcal{I}_u} \sum_{j \in \mathcal{I}_v} w_{ui} \cdot w_{vj} \cdot s(i, j), \quad (3)$$

where w_{ui} is the normalized weight of the link between u and i . Like *SimRank*, this method also has some limitations. First, since link weights are simply multiplied it may not be possible to compare the agreement between the ratings made by two users on similar items, especially if these ratings are non-numerical. Also, this method does not allow one to integrate prior knowledge on the similarity values, for instance, obtained by comparing the content of items.

2.3 Contributions

This paper makes the following contributions:

1. It describes a novel approach to compute similarities that extends the *SimRank* algorithm and its extensions in two important ways:
 - (a) It uses an arbitrary function to compare the agreement between link weights, which allows the use of non-numerical ratings.
 - (b) It provides an elegant way to integrate prior information on the similarity values directly in the computations.
2. Unlike similarity measures based on correlation which only use the ratings on common items, this approach considers all the available ratings, allowing it to compute similarities between users that have rated different items, thereby reducing the sensitivity to sparse data.
3. It presents a first comprehensive experimental evaluation of a *SimRank*-based method on the problem of predicting new ratings.

3 A Novel Approach

3.1 The General Formulation

Consider the task of evaluating the similarity $s(u, v)$ between two users u and v . A simple approach, used in several item recommendation systems, is to compute $s(u, v)$ as the correlation between the ratings given by u and v on common items. Besides being limited to numerical ratings, this approach has another significant problem: similarities can only be evaluated for users that have rated common items, and the correlation values are only significant if there is a sufficient number of common items. For these reasons, the correlation approach gives poor results when the rating data is sparse.

As in *SimRank*, our approach overcomes these limitations by using all the ratings given by u and v , not only those given to common items. Thus, we evaluate the similarity between users u and v as the average rating agreement for all pairs of rated items, weighted by the similarity of these items:

$$s(u, v) = \frac{1}{Z_{uv}} \sum_{i \in \mathcal{I}_u} \sum_{j \in \mathcal{I}_v} s(i, j) k(r_{ui}, r_{vj}), \quad (4)$$

where k is a function that evaluates the agreement between two (possibly non-numerical) ratings, and Z_{uv} is a normalization constant, for instance, $Z_{uv} = |\mathcal{I}_u||\mathcal{I}_v|$. Examples of agreement function k for *numerical* ratings are the *Radial Basis Function* (RBF) Gaussian kernel

$$k_{\text{RBF}}(r_{ui}, r_{vj}) = \exp\{-(r_{ui} - r_{vj})^2/\gamma^2\}, \tag{5}$$

where γ controls the width of the kernel, and the *Correlation* kernel

$$k_{\text{Cor}}(r_{ui}, r_{vj}) = \frac{(r_{ui} - \bar{r}_u)(r_{vj} - \bar{r}_v)}{\sigma_u \sigma_v}, \tag{6}$$

\bar{r}_u and σ_u being the mean and standard deviation of the ratings given by u . Note that k does not need to be semi-definite positive (SDP), and the term *kernel* is used in a more general way to represent a function measuring similarity.

A benefit of this formulation is that the agreement between two ratings is abstracted in function k , which can be tailored to model specific characteristics or constraints of the system, as well as to measure the agreement between any rating types. Moreover, this formulation can be easily extended to include prior information on the similarity between users u and v , obtained, for example, by comparing their profiles (*gender, age, etc.*). Denote $\hat{s}(u, v)$ the *a priori* similarity capturing this information, (4) can be extended to include $\hat{s}(u, v)$ as

$$s(u, v) = (1 - \alpha)\hat{s}(u, v) + \frac{\alpha}{Z_{uv}} \sum_{i \in \mathcal{I}_u} \sum_{j \in \mathcal{I}_v} s(i, j) k(r_{ui}, r_{vj}), \tag{7}$$

where $\alpha \in [0, 1]$ controls the importance of the *a priori* similarity in the computation. Likewise, the similarity $s(i, j)$ between two items $i, j \in \mathcal{I}$ can be modeled as

$$s(i, j) = (1 - \alpha)\hat{s}(i, j) + \frac{\alpha}{Z_{ij}} \sum_{u \in \mathcal{U}_i} \sum_{v \in \mathcal{U}_j} s(u, v) k(r_{ui}, r_{vj}), \tag{8}$$

where $\hat{s}(i, j)$ models prior knowledge on the similarity between i and j , for instance, their content similarity, and Z_{ij} has the same role as Z_{uv} .

3.2 Modeling Similarities as a Linear System

The relations between similarity values, as defined by equations (7) and (8), form a linear system which can be described using a matricial notation. Denote the user and item similarities as vectors $\mathbf{x} \in \mathbb{R}^{|\mathcal{U}|^2}$ and $\mathbf{y} \in \mathbb{R}^{|\mathcal{I}|^2}$ such that each pair of users u, v is mapped to a unique element $\mathbf{x}_{(uv)} = s(u, v)$, and each pair of items i, j maps to a unique element $\mathbf{y}_{(ij)} = s(i, j)$. Also, let $\mathbf{c} \in \mathbb{R}^{|\mathcal{U}|^2}$ and $\mathbf{d} \in \mathbb{R}^{|\mathcal{I}|^2}$ be vectors such that $\mathbf{c}_{(uv)} = \hat{s}(u, v)$ and $\mathbf{d}_{(ij)} = \hat{s}(i, j)$. Moreover, define A as the $(|\mathcal{U}|^2 \times |\mathcal{I}|^2)$ matrix such that $A_{(uv)(ij)} = \frac{1}{Z_{uv}}k(r_{ui}, r_{vj})$, if $i \in \mathcal{I}_u$ and $j \in \mathcal{I}_v$, and $A_{(uv)(ij)} = 0$ otherwise. Likewise, let B is a $(|\mathcal{I}|^2 \times |\mathcal{U}|^2)$ matrix such that $B_{(ij)(uv)} = \frac{1}{Z_{ij}}k(r_{ui}, r_{vj})$ if $u \in \mathcal{U}_i$ and $v \in \mathcal{U}_j$, and $B_{(ij)(uv)} = 0$ otherwise.

The linear system formed of equations (7) and (8) can thus be written in matrix form as

$$\begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} = (1 - \alpha) \begin{pmatrix} \mathbf{c} \\ \mathbf{d} \end{pmatrix} + \alpha \begin{pmatrix} 0 & A \\ B & 0 \end{pmatrix} \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix}, \quad (9)$$

and has the following solution:

$$\begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} = (1 - \alpha) \begin{pmatrix} I & -\alpha A \\ -\alpha B & I \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{c} \\ \mathbf{d} \end{pmatrix} = (1 - \alpha) \begin{pmatrix} R^{-1} & \alpha AS^{-1} \\ \alpha BR^{-1} & S^{-1} \end{pmatrix} \begin{pmatrix} \mathbf{c} \\ \mathbf{d} \end{pmatrix}, \quad (10)$$

where $R = (I - \alpha^2 AB)$ and $S = (I - \alpha^2 BA)$.

3.3 Computing the Similarities

Although A and B may be very sparse matrices, their large size can render difficult the direct computation of R^{-1} and S^{-1} . A more efficient approach consists in using an iterative method based on the *von Neumann series* expansion of these matrices [11,13]:

$$R^{-1} = \sum_{n=0}^{\infty} (\alpha^2 AB)^n \quad \text{and} \quad S^{-1} = \sum_{n=0}^{\infty} (\alpha^2 BA)^n.$$

The solution for \mathbf{x} can therefore be expressed as

$$\mathbf{x} = (1 - \alpha) \left(\sum_{n=0}^{\infty} (\alpha^2 AB)^n \mathbf{c} + \alpha A \sum_{n=0}^{\infty} (\alpha^2 BA)^n \mathbf{d} \right) = \left(\sum_{n=0}^{\infty} (\alpha^2 AB)^n \right) \mathbf{p}. \quad (11)$$

where $\mathbf{p} = (1 - \alpha)(\mathbf{c} + \alpha A\mathbf{d})$. Using the same approach, \mathbf{y} is obtained as

$$\mathbf{y} = \left(\sum_{n=0}^{\infty} (\alpha^2 BA)^n \right) \mathbf{q}, \quad (12)$$

where $\mathbf{q} = (1 - \alpha)(\alpha B\mathbf{c} + \mathbf{d})$.

This new formulation leads to a simple method to compute \mathbf{x} and \mathbf{y} . Since a similar approach can be used for \mathbf{y} , we limit our presentation to the computation of \mathbf{x} . First, the method initializes \mathbf{x} to the *null* vector and initializes a temporary vector \mathbf{w} to \mathbf{p} . Then, the following two steps are repeated until convergence or a maximum number of iterations is reached:

1. Update the similarities vector: $\mathbf{x} \leftarrow \mathbf{x} + \mathbf{w}$,
2. Update the temporary vector: $\mathbf{w} \leftarrow \alpha^2 AB\mathbf{w}$.

Theorem 1. Denote by λ_{\max} the largest eigenvalue of matrix AB , also known as its spectral radius. The iterative method presented above converges if $\alpha^2 |\lambda_{\max}| < 1$.

Proof. Let $X\Lambda X^{-1}$ be the eigen-decomposition of matrix AB . At the n -th iteration, we have

$$\|(\alpha^2 AB)^n\| = \|X(\alpha^2 A)^n X^{-1}\| \leq \|X\| \cdot \sqrt{\sum_i (\alpha^2 \lambda_i)^{2n}} \cdot \|X^{-1}\|.$$

If $\alpha^2 |\lambda_{\max}| < 1$ then $\|(\alpha^2 AB)^n\|$ will converge to 0 as n approaches infinity. As a consequence, \mathbf{x} will converge to a fixed value.

To analyze the complexity of this approach, as observed in most recommender systems, we suppose the number of ratings given by any user to be bounded by a constant m independent of the number of items. Since $A_{(uv)(ij)}$ is non-zero only if $i \in \mathcal{I}_u$ and $j \in \mathcal{I}_v$, assuming an even distribution of ratings among the users and items, the expected number of non-zero values in A is given by

$$\frac{|\mathcal{U}|^2 |\mathcal{I}|^2}{2} \times \left(\frac{m}{|\mathcal{I}|}\right)^2 = \frac{|\mathcal{U}|^2 m^2}{2} \in O(|\mathcal{U}|^2).$$

Likewise, we find the expected number of non-zero elements of B to be in $O(|\mathcal{U}|^2)$. Moreover, because the method has to store the non-zero values of A and B , as well as the values of possibly dense vectors \mathbf{x} and \mathbf{p} , the expected space complexity of the method is $O(|\mathcal{U}|^2)$. For the time complexity, the dominant operations are the two matrix multiplications: $B\mathbf{w} = \mathbf{w}'$ and $A\mathbf{w}'$. Since the complexity of these operations is proportional to the number of non-zero elements in the multiplying matrices, the total expected time complexity of the method is $O(n_{\max} |\mathcal{U}|^2)$, where n_{\max} is the maximum number of iterations made by the method. While n_{\max} largely depends on the normalization constants Z_{uv} and Z_{ij} , as well as on the link agreement function k , in our experiments, the method would normally take 5 to 10 iterations to converge.

3.4 Solving without Prior Information

Although it is always possible to use default values for \mathbf{c} and \mathbf{d} , for instance $c_{(uv)} = 1$ if $u = v$ and 0 otherwise, the approach proposed in this paper could also be used without such information. The following theorem explains how this can be done.

Theorem 2. *Let G be a directed weighted bipartite graph constructed such that each pair of users u, v corresponds to a node (uv) from the first set of nodes, each pair of items i, j is a node (ij) from the second set, and whose adjacency matrix is*

$$adj(G) = \begin{pmatrix} 0 & A \\ B & 0 \end{pmatrix}.$$

If $\alpha = 1$, A, B are non-negative matrices and G is connected, then vectors \mathbf{x} and \mathbf{y} correspond, respectively, to the unique eigenvectors of matrices AB and BA associated with the largest eigenvalue of these matrices. Moreover, these eigenvectors can be computed using a power iteration method [7].

Proof. Suppose we constrain \mathbf{x} and \mathbf{y} to a specific length, for instance $\|\mathbf{x}\| = \|\mathbf{y}\| = 1$, then equations (7) and (8) can be expressed as $\mathbf{x} = \frac{1}{\omega} A\mathbf{y}$ and $\mathbf{y} = \frac{1}{\sigma} B\mathbf{x}$,

where ω and σ are normalization constants. Inserting the second one into the first, we get $(\sigma\omega)\mathbf{x} = AB\mathbf{x}$ and, thus, \mathbf{x} is an eigenvector of AB corresponding to the eigenvalue $\lambda = \sigma\omega$. Likewise, \mathbf{y} is an eigenvector of BA corresponding to the *same* eigenvalue.

Furthermore, since A and B are non-negative, so are matrices AB and BA . Also, because G is connected, and since $A_{(uv)(ij)} > 0$ if and only if $B_{(ij)(uv)} > 0$, G is also strongly connected. Consequently the graph with node set \mathcal{U}^2 and adjacency matrix AB , and the graph with node set \mathcal{I}^2 and adjacency matrix BA are also strongly connected. This, in turn, is equivalent to saying that AB and BA are irreducible matrices. Finally, since AB and BA are square, non-negative, irreducible matrices, by the Perron-Frobenius theorem on non-negative matrices, the eigenspace corresponding to the eigenvalue λ_{\max} of largest magnitude is of dimension one and contains an eigenvector whose components are all positive. Running two parallel *power iteration* methods on matrices AB and BA will therefore converge to the unique positive eigenvectors of AB and BA , associated to λ_{\max} [7]. The convergence of this method is geometric with respect to $\frac{|\lambda'_{\max}|}{|\lambda_{\max}|} < 1$, where λ'_{\max} is the eigenvalue of second largest magnitude.

Following Theorem 2, the similarity values can be computed by repeating the following two steps until convergence:

1. Update the *normalized* user similarities: $\mathbf{x} \leftarrow A\mathbf{y} / \|A\mathbf{y}\|$,
2. Update the *normalized* item similarities: $\mathbf{y} \leftarrow B\mathbf{x} / \|B\mathbf{x}\|$.

Once again, this approach usually converges within a few iterations and the complexity of each iteration is reduced by the fact that matrices A and B are normally quite sparse.

4 Experimental Evaluation

In this section, we evaluate our approach on the task of predicting the ratings of users for movies and jokes. As it is tailored to compute similarities in sparse data, and not specifically to predict ratings, it should be recognized that our approach is not directly comparable with state-of-the-art methods for this task. Yet, evaluating our approach on this problem still provides valuable information, as it allows us to measure the quality of its computed similarities. To this end, we compare the similarities obtained by our method with those computed with correlation-based and SVD methods, in the nearest-neighbor prediction of ratings. Since all three types of similarities use the same approach to predict ratings, more accurate predictions indicate more relevant similarity values.

4.1 Tested Methods

In our experiments we compared three methods to compute similarities. The first one, called ESR (*Enhanced SimRank*), is the approach described in this paper. For these experiments, we used $Z_{uv} = |\mathcal{I}_u||\mathcal{I}_v|$ and $Z_{ij} = |\mathcal{U}_i||\mathcal{U}_j|$ as

normalization constants and the Gaussian RBF kernel of (5) with $\gamma = 0.05$ as the rating agreement function. However, this kernel was used in a slightly different way for matrices A and B . Thus, for A , the kernel was computed on the *normalized* ratings $(r_{ui} - \bar{r}_u)/(r_{\max} - r_{\min})$, where \bar{r}_u is the average rating given by user u and r_{\min} , r_{\max} are the minimum and maximum values of the rating range. For B , however, the kernel was computed on ratings normalized as $(r_{ui} - \bar{r}_i)/(r_{\max} - r_{\min})$, where \bar{r}_i is the average rating given to item i . Finally, we used $\alpha = 0.95$ as the blending factor and defined the *a priori* similarity values as

$$\hat{s}(u, v) \text{ (resp. } \hat{s}(i, j)) = \begin{cases} 1.0, & \text{if } u = v \text{ (resp. } i = j), \\ 0.1, & \text{otherwise.} \end{cases}$$

These parameter values were selected based on cross-validation.

The second method, denoted by PCC, is the Pearson correlation similarity. Following the literature (e.g., see [16]), we computed user similarities as

$$s(u, v) = \frac{\sum_{i \in \mathcal{I}_{uv}} (r_{ui} - \bar{r}_u)(r_{vi} - \bar{r}_v)}{\sqrt{\sum_{i \in \mathcal{I}_{uv}} (r_{ui} - \bar{r}_u)^2 \sum_{i \in \mathcal{I}_{uv}} (r_{vi} - \bar{r}_v)^2}}. \tag{13}$$

and the item similarities as

$$s(i, j) = \frac{\sum_{u \in \mathcal{U}_{ij}} (r_{ui} - \bar{r}_i)(r_{uj} - \bar{r}_j)}{\sqrt{\sum_{u \in \mathcal{U}_{ij}} (r_{ui} - \bar{r}_i)^2 \sum_{u \in \mathcal{U}_{ij}} (r_{uj} - \bar{r}_j)^2}}. \tag{14}$$

Finally, the third method, called SVD, is based on the decomposition of the rating matrix. Like the approach described in [17], we represented each user u by a vector $\mathbf{p}_u \in \mathbb{R}^f$ and each item by a vector $\mathbf{q}_i \in \mathbb{R}^f$, where f is the dimensionality of the latent space. Vectors \mathbf{p}_u and \mathbf{q}_i were then learned from the data by solving the following problem:

$$\min_{\mathbf{p}, \mathbf{q}} \sum_{z_{ui} \in \mathcal{D}} (z_{ui} - \mathbf{p}_u^\top \mathbf{q}_i)^2 \quad \text{s.t.} \quad \|\mathbf{p}_u\| = \|\mathbf{q}_i\| = 1, \quad \forall u \in \mathcal{U}, \forall i \in \mathcal{I}, \tag{15}$$

where $z_{ui} = (r_{ui} - \bar{r}_i)/(r_{\max} - r_{\min})$. This problem corresponds to finding, for each user u and item i , coordinates on the surface of the f -dimensional unit sphere such that u will give a high rating to i if their coordinates are close together on the surface. If two users u and v are nearby on the surface, then they will give similar ratings to the same items, and, thus, the similarity between these users can be computed as $s(u, v) = \mathbf{p}_u^\top \mathbf{p}_v$. Likewise, the similarity between two items i and j can be obtained as $s(i, j) = \mathbf{q}_i^\top \mathbf{q}_j$. Based on cross-validation, we have used $f = 50$ in our experiments.

The similarities obtained with these three methods were used to predict ratings r_{ui} in two different ways. In the first approach, called *user-based* prediction [12], the K nearest-neighbors of u that have rated i , denoted by $\mathcal{N}_i(u)$, are

found with the users similarities. The ratings of these users for i are then used to predict r_{ui} as

$$\hat{r}_{ui} = \bar{r}_u + \sum_{v \in \mathcal{N}_i(u)} s(u, v) \cdot (r_{vi} - \bar{r}_v) / \sum_{v \in \mathcal{N}_i(u)} |s(u, v)|. \quad (16)$$

The second approach, known as *item-based* prediction [4], instead uses the item similarities to find the K nearest-neighbors of item i that have been rated by u , denoted by $\mathcal{N}_u(i)$, and predicts ratings as

$$\hat{r}_{ui} = \bar{r}_i + \sum_{j \in \mathcal{N}_u(i)} s(i, j) \cdot (r_{uj} - \bar{r}_j) / \sum_{j \in \mathcal{N}_u(i)} |s(i, j)|. \quad (17)$$

In the experiments presented in this section, we used $K = 50$ as the number of nearest-neighbors considered in the prediction.

4.2 Benchmark Datasets

We tested the prediction approaches on three different real-life datasets, *MovieLens*¹, *Netflix*² and *Jester*³, coming from systems recommending movies and jokes. The properties of these datasets are given in Table 1. Compared to the other two, the *Jester* dataset is particularly dense, with 410,000 ratings per joke on average. This dataset also differs from the others by the fact that its rating scale is continuous.

Table 1. Properties of the benchmark datasets

Dataset	Type	Nb. users	Nb. items	Nb. ratings	Rating range
<i>MovieLens</i>	Movies	6,040	3,952	1 M	{1, 2, 3, 4, 5}
<i>Netflix</i>	Movies	480,189	17,770	100 M	{1, 2, 3, 4, 5}
<i>Jester</i>	Jokes	72,421	100	4.1 M	[-10, 10]

To generate datasets of various sparsity levels, we randomly selected 5,000 users from the *Netflix* and *Jester* datasets, and discarded the ratings that were not made by these users (the ratings of the *MovieLens* dataset were all kept). Then, for all three datasets, we sub-sampled the ratings of the remaining users by randomly selecting a user $u \in \mathcal{U}$ with a probability proportional to $|\mathcal{I}_u|$ and randomly removed one of its ratings from \mathcal{I}_u . We repeated this sub-sampling process until $|\mathcal{U}| \times \rho_u$ ratings were left, where ρ_u is the desired average number of ratings per user. To avoid having users with too few ratings, however, we allowed removing a rating from user u only if $|\mathcal{I}_u| > 0.5 \times \rho_u$. Using an average number of ratings ρ_u of 5, 10, 15 and 20, we obtained with this approach four subsets for each of the *MovieLens*, *Netflix* and *Jester* datasets. Note that, although the *MovieLens* and *Netflix* datasets contain information on the users and movies,

¹ <http://www.grouplens.org/>

² <http://www.netflixprize.com/>

³ <http://www.ieor.berkeley.edu/~goldberg/jester-data/>

as well as timestamps indicating when the ratings were made, we did not take such information into account in these experiments.

To assess the performance of these strategies, we used a 10-fold cross-validation scheme, where the dataset \mathcal{D} was randomly split in 10 equal sized subsets \mathcal{D}_k , $k = 1, \dots, 10$. For each k , we used $\bigcup_{l \neq k} \mathcal{D}_l$ to compute the user and item similarities (training phase) and then evaluated the *Mean Absolute Error* (MAE) and the *Root Mean Squared Error* (RMSE) on subset \mathcal{D}_k . The reported error values were taken as the mean errors over all 10 subsets.

MOVIELENS DATA SUBSETS

ρ_u	Result	USER-BASED PREDICTION			ITEM-BASED PREDICTION		
		PCC	SVD	ESR	PCC	SVD	ESR
5	MAE	0.934 (.011)	0.870 (.014)	0.854 (.011)	0.857 (.018)	0.883 (.018)	0.811 (.011)
	RMSE	1.236 (.012)	1.156 (.017)	1.128 (.012)	1.134 (.017)	1.162 (.017)	1.076 (.012)
	#NN	0.7	24.5	24.5	0.6	4.2	4.2
10	MAE	0.897 (.009)	0.798 (.010)	0.783 (.009)	0.860 (.004)	0.832 (.008)	0.754 (.010)
	RMSE	1.170 (.009)	1.060 (.010)	1.036 (.011)	1.133 (.007)	1.096 (.011)	1.005 (.012)
	#NN	8.2	34.1	34.1	3.9	10.4	10.4
15	MAE	0.841 (.008)	0.776 (.006)	0.762 (.010)	0.818 (.008)	0.803 (.007)	0.735 (.008)
	RMSE	1.104 (.010)	1.033 (.009)	1.006 (.010)	1.079 (.010)	1.061 (.007)	0.970 (.010)
	#NN	19.7	38.7	38.7	8.1	15.6	15.6
20	MAE	0.807 (.007)	0.773 (.005)	0.753 (.006)	0.785 (.007)	0.786 (.010)	0.723 (.006)
	RMSE	1.063 (.006)	1.027 (.007)	0.991 (.005)	1.039 (.007)	1.038 (.011)	0.965 (.007)
	#NN	29.3	41.4	41.4	13.3	21.1	21.1

NETFLIX DATA SUBSETS

ρ_u	Result	USER-BASED PREDICTION			ITEM-BASED PREDICTION		
		PCC	SVD	ESR	PCC	SVD	ESR
5	MAE	0.914 (.016)	0.896 (.020)	0.877 (.020)	0.929 (.012)	0.960 (.019)	0.881 (.011)
	RMSE	1.216 (.021)	1.190 (.021)	1.166 (.022)	1.220 (.015)	1.247 (.021)	1.164 (.016)
	#NN	0.5	18.7	18.7	0.4	4.3	4.3
10	MAE	0.890 (.013)	0.845 (.007)	0.811 (.011)	0.920 (.010)	0.894 (.013)	0.819 (.007)
	RMSE	1.170 (.014)	1.117 (.007)	1.081 (.012)	1.213 (.011)	1.171 (.012)	1.086 (.010)
	#NN	5.2	26.9	26.9	2.5	10.6	10.6
15	MAE	0.867 (.008)	0.832 (.011)	0.790 (.011)	0.893 (.010)	0.867 (.008)	0.792 (.011)
	RMSE	1.134 (.011)	1.102 (.011)	1.055 (.011)	1.175 (.011)	1.137 (.010)	1.058 (.013)
	#NN	12.9	31.0	31.0	5.5	15.9	15.9
20	MAE	0.839 (.007)	0.824 (.007)	0.776 (.008)	0.860 (.006)	0.848 (.005)	0.776 (.005)
	RMSE	1.103 (.009)	1.090 (.007)	1.037 (.009)	1.138 (.005)	1.114 (.008)	1.039 (.006)
	#NN	20.5	33.7	33.7	9.2	21.4	21.4

JESTER DATA SUBSETS

ρ_u	Result	USER-BASED PREDICTION			ITEM-BASED PREDICTION		
		PCC	SVD	ESR	PCC	SVD	ESR
5	MAE	4.076 (.072)	3.940 (.050)	3.896 (.064)	4.060 (.047)	4.714 (.083)	3.809 (.058)
	RMSE	5.194 (.081)	5.063 (.079)	5.017 (.073)	5.212 (.065)	5.953 (.109)	4.891 (.069)
	#NN	39.5	50.0	50.0	4.1	4.1	4.1
10	MAE	3.710 (.059)	3.675 (.053)	3.655 (.062)	3.588 (.052)	4.345 (.042)	3.603 (.055)
	RMSE	4.695 (.067)	4.702 (.054)	4.651 (.074)	4.587 (.069)	5.410 (.041)	4.592 (.067)
	#NN	50.0	50.0	50.0	9.1	9.1	9.1
15	MAE	3.665 (.035)	3.571 (.029)	3.581 (.038)	3.476 (.039)	4.193 (.038)	3.539 (.039)
	RMSE	4.617 (.049)	4.567 (.032)	4.538 (.049)	4.434 (.050)	5.184 (.038)	4.493 (.051)
	#NN	50.0	50.0	50.0	13.9	13.9	13.9
20	MAE	3.634 (.018)	3.505 (.019)	3.541 (.015)	3.431 (.020)	4.143 (.031)	3.511 (.018)
	RMSE	4.568 (.027)	4.490 (.024)	4.480 (.025)	4.365 (.028)	5.105 (.032)	4.444 (.027)
	#NN	50.0	50.0	50.0	18.9	18.9	18.9

Fig. 2. Average MAE and RMSE (and corresponding standard deviation) obtained for the *MovieLens*, *Netflix* and *Jester* data subsets, with an average number of ratings per user $\rho_u \in \{5, 10, 15, 20\}$. #NN gives the average number of neighbors used in the predictions.

4.3 Prediction Results

Figure 2 presents the results for the six rating prediction methods on the *MovieLens*, *Netflix* and *Jester* data subsets. The lower the MAE and RMSE values, the more accurate are the methods at predicting ratings. Moreover, the #NN values give the average number of neighbors used in the predictions. A low value indicates that a significant portion of the user or item similarities are equal to zero, due to data sparsity.

From these results, we can see that the similarity values obtained by our method leads to more accurate predictions than those of the SVD method, even though these predictions were made with the same number of neighbors. Moreover, compared to PCC, our method also leads to better results on the sparser datasets *MovieLens* and *Netflix*. However, in the denser *Jester* dataset, PCC similarities produce more accurate predictions for $\rho_u = 15$ and $\rho_u = 20$. Even though we have used only a sub-sample of the ratings, one should note that the *Jester* data subsets tested in our experiments are still very dense. Thus, for $\rho_u = 15$, users still have rated on average 15% of the jokes. Nevertheless, the result of this experiments seem to indicate that our method provides better similarity values when the data is sparse, but correlation based approaches might be superior when a large number of ratings is available.

5 Summary and Future Works

This paper presented a novel approach to compute similarities. Like *SimRank*, our approach uses a formulation that associates similarities between linked objects of two different sets. However, our approach also allows one to model the agreement between link weights using any desired function and provides an elegant way to integrate prior information on the similarity values directly in the computations.

To illustrate its usefulness, we have described how this approach can be used to evaluate the similarities between the users or the items of a recommender system, based on the ratings of users on items. In contrast to the traditional methods using rating correlation, our approach has the benefit of considering all the available ratings made by two users, making possible the computation of similarities between users that have rated different items. Also, as opposed to more recent recommendation methods, this approach is not limited to numerical ratings and provides a simple way to integrate information on item content or user profile similarity. Finally, experiments conducted on the problem of predicting new ratings on three different real-life datasets have shown the similarities obtained with our approach to lead to more accurate predictions than those obtained by two other methods based on Pearson correlation and on SVD, when the data is sparse.

In future works, we would like to deeper investigate the impact of using prior knowledge on the similarities, for instance, obtained from user profiles and item content. Moreover, we also consider defining and evaluating other types of rating agreement functions, in particular, in the setting where ratings are non-numerical.

References

1. Antonellis, L., Molina, H.G., Chang, C.C.: Simrank++: query rewriting through link analysis of the click graph. *Proceedings of the VLDB Endowment* 1(1), 408–421 (2008)
2. Bell, R.M., Koren, Y., Volinsky, C.: Modeling relationships at multiple scales to improve accuracy of large recommender systems. In: *KDD 2007: Proc. of the 13th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, pp. 95–104. ACM, New York (2007)
3. Bell, R.M., Koren, Y.: Scalable collaborative filtering with jointly derived neighborhood interpolation weights. In: *ICDM 2007: Proc. of the 2007 Seventh IEEE Int. Conf. on Data Mining*, pp. 43–52. IEEE Computer Society, Washington (2007)
4. Deshpande, M., Karypis, G.: Item-based top-N recommendation algorithms. *ACM Transaction on Information Systems* 22(1), 143–177 (2004)
5. Fouss, F., Renders, J.-M., Pirotte, A., Saerens, M.: Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation. *IEEE Transactions on Knowledge and Data Engineering* 19(3), 355–369 (2007)
6. Goldberg, K., Roeder, T., Gupta, D., Perkins, C.: Eigentaste: A constant time collaborative filtering algorithm. *Information Retrieval* 4(2), 133–151 (2001)
7. Golub, G.H., Van Loan, C.F.: *Matrix computations*, 3rd edn. Johns Hopkins University Press, Baltimore (1996)
8. Gori, M., Pucci, A.: Itemrank: a random-walk based scoring algorithm for recommender engines. In: *Proc. of the 2007 IJCAI Conf.*, pp. 2766–2771 (2007)
9. Jeh, G., Widom, J.: Simrank: a measure of structural-context similarity. In: *KDD 2002: Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 538–543. ACM, New York (2002)
10. Kim, B.M., Li, Q., Park, C.S., Kim, S.G., Kim, J.Y.: A new approach for combining content-based and collaborative filters. *Journal of Intelligent Information Systems* 27(1), 79–91 (2006)
11. Kondor, R.I., Lafferty, J.D.: Diffusion kernels on graphs and other discrete input spaces. In: *ICML 2002: Proc. of the Nineteenth Int. Conf. on Machine Learning*, pp. 315–322. Morgan Kaufmann Publishers Inc., San Francisco (2002)
12. Konstan, J.A., Miller, B.N., Maltz, D., Herlocker, J.L., Gordon, L.R., Riedl, J.: GroupLens: applying collaborative filtering to usenet news. *Communications of the ACM* 40(3), 77–87 (1997)
13. Kunegis, J., Lommatzsch, A., Bauckhage, C.: Alternative similarity functions for graph kernels. In: *Proc. of the Int. Conf. on Pattern Recognition* (2008)
14. Li, J., Zaiane, O.R.: Combining usage, content, and structure data to improve Web site recommendation. In: *Bauknecht, K., Bichler, M., Pröll, B. (eds.) EC-Web 2004. LNCS, vol. 3182*, pp. 305–315. Springer, Heidelberg (2004)
15. Luo, H., Niu, C., Shen, R., Ullrich, C.: A collaborative filtering framework based on both local user similarity and global user similarity. *Machine Learning* 72(3), 231–245 (2008)
16. Sarwar, B., Karypis, G., Konstan, J., Reidl, J.: Item-based collaborative filtering recommendation algorithms. In: *WWW 2001: Proc. of the 10th Int. Conf. on World Wide Web*, pp. 285–295. ACM, New York (2001)
17. Takács, G., Pilászy, I., Németh, B., Tikk, D.: Major components of the gravity recommendation system. *SIGKDD Exploration Newsletter* 9(2), 80–83 (2007)
18. Yildirim, H., Krishnamoorthy, M.S.: A random walk method for alleviating the sparsity problem in collaborative filtering. In: *RecSys 2008: Proc. of the 2008 ACM Conf. on Recommender systems*, pp. 131–138. ACM, New York (2008)

Multiobjective Optimization Approach for Named Entity Recognition

Asif Ekbal¹, Sriparna Saha^{2,*}, and Christoph S. Garbe²

¹ Department of Computational Linguistics, Heidelberg University, Germany
ekbal@cl.uni-heidelberg.de, asif.ekbal@gmail.com

² Image Processing and Modeling, Interdisciplinary Center for Scientific Computing (IWR), Heidelberg University, Heidelberg, Germany
sriparna.saha@iwr.uni-heidelberg.de, Christoph.Garbe@uni-heidelberg.de

Abstract. In this paper, we propose a multiobjective optimization (MOO) based technique to determine the appropriate weight of voting for each class in each classifier for Named Entity Recognition (NER). Our underlying assumption is that reliability of predictions of each classifier differs among the various named entity (NE) classes. Thus, it is necessary to quantify the amount of voting for each class in a particular classifier. We use Maximum Entropy (ME) as the base to generate a number of classifiers depending upon the various feature representations. The proposed algorithm is evaluated for a resource-constrained language like Bengali that yield the overall recall, precision and F-measure values of 79.98%, 82.24% and 81.10%, respectively. Experiments also show that the classifier ensemble identified by the proposed multiobjective based technique outperforms all the individual classifiers, three different conventional *baseline* ensembles and an existing single objective optimization based approach.

1 Introduction

Named Entity Recognition (NER) is an important pipelined module in many Natural Language Processing (NLP) application areas such as information extraction [1], machine translation [2], question answering [3] and automatic summarization [4] etc. Named Entity (NE) identification in Indian languages in general and Bengali in particular is more difficult and challenging compared to English, most of the European languages and some of the Asian languages such as Chinese, Japanese and Korean. The difficulties lie with some of the facts such as: (i). missing of capitalization information, (ii). appearance of NEs in the dictionary with some other specific meanings, (iii). free word order nature of the languages and (iv). resource-constrained environment, i.e., non-availability of corpora, annotated corpora, name dictionaries, good morphological analyzers, part of speech (POS) taggers etc. in the required measure. Thus, developing reasonably high accurate NE taggers for such resource-poor languages is a big challenge.

* The first two authors are the joint first authors.

In the area of machine learning, the concept of combining (or, ensembling) classifiers has drawn much attention to the researchers during the last few years with the aim of achieving better performance in comparison to the individual classifiers, that could be of homogeneous or heterogenous types. Feature selection is also a very crucial issue in machine learning. In the present work, we assume that rather than searching for the best-fitting feature set, ensembling several homogenous NER systems where each one is based on different feature representation could be more effective. But, the selection of appropriate subset of classifiers is very crucial. Moreover, all the classifiers are not good to detect all types of NE classes. For ensembling the outputs of all classifiers, either majority voting or weighted voting is used. In case of weighted voting, weights should vary among the various NE classes in each classifier. The weight of a particular classifier should be high for that particular NE class for which it performs good. Otherwise, weights should be low for the NE classes for which its outputs are not very reliable. Some single objective optimization techniques like genetic algorithm (GA) [5] can be used to determine the appropriate weight combinations per classifier [6]. This single objective optimization technique can only optimize a single quality measure, e.g., recall, precision or F-measure at a time. But, sometimes a single measure cannot capture the quality of a good ensembling reliably. A good weighted vote based ensemble should have its all the parameters optimized simultaneously. In order to achieve this, we use multiobjective optimization (MOO) [7] that is capable of simultaneously optimizing more than one classification quality measures. Experimental results also justify that MOO performs superior compared to the single objective optimization for NER. We use ME framework as a base classifier. Depending on the various combinations of the available features, different versions of this classifier are made. These features are language independent in nature, and can be derived for almost all the languages with a very little effort.

The proposed technique is evaluated for a resource constrained language like Bengali. In terms of native speakers, Bengali is the *fifth* popular language in the world, *second* in India and the *national* language in Bangladesh. We manually annotate approximately 250K wordforms that were randomly selected from a portion of the Bengali news corpus [8], developed from the archive of leading newspaper available in the web. In addition, we also use the IJCNLP-08 NER on South and South East Asian Languages (NERSSEAL)¹ Shared Task data of around 100K wordforms. Evaluation results of our proposed method yield the recall, precision and F-measure values of 79.98%, 82.24% and 81.10%, respectively. Results also show that the classifier ensemble identified by our proposed technique outperforms all the individual classifiers, three different *baseline* ensembles and a single objective optimization based approach [6].

In the literature, there exists some works related to NER that made use of classifier combination techniques. For example, Florian et al. [9] reported a system by combining four diverse classifiers that exhibited best performance in the CoNLL-2003 shared task [10]. In Indian languages, the classifier combination

¹ <http://lrec.iit.ac.in/ner-ssea-08>

technique for NER has been reported in Ekbal and Bandyopadhyay [11] for Bengali. But, these two works are based on the multiple heterogeneous classifiers, and used more complex experimental set up along with the domain dependent resources. In contrast, our system (i). is based only on the ME framework, (ii). makes use of a small set of features that can be very easily obtained for many languages and (iii). does not make use of domain dependent resources, but still achieves the state-of-the-art performance.

The key contributions of our work are listed below:

1. A MOO based technique is proposed for selecting the best weights to form a classifier ensemble². We tried to establish that such ensemble is capable to increase the classification quality by a large margin compared to the conventional ensemble methods.
2. ME is used as a test classifier due to its less computational overhead. However, the proposed method will work for any set of classifiers, i.e. either homogeneous or heterogeneous. The proposed technique is very general and its performance may further improve depending upon the choice and/or the number of classifiers as well as the use of more complex features.
3. The proposed technique can be replicated for any resource-poor language very easily due to its language independent nature.
4. The proposed technique is applicable for any type of classification problems like NER, POS-tagging, question-answering etc. To the best of our knowledge, use of MOO to select appropriate weights for voting is a novel contribution.
5. Note, that our work proposes a novel way of ensembling the available classifiers. Performance of the existing works, that are based on ensemble techniques (e.g., [11], [9] etc.), can be further improved with our proposed algorithm.
6. Another important motivation of MOO based technique is to provide the users a set of alternative solutions with high recall values or solutions with high precision values or solutions with moderate recall and precision values. Depending upon the nature of problems or the requirement of the users, appropriate solutions can be selected.

2 Problem Formulation

In this section, we formulate the weighted vote based classifier ensemble problem under the MOO framework. Let, the N number of available classifiers be denoted by C_1, \dots, C_N and $\mathcal{A} = \{C_i : i = 1; N\}$. Suppose, there are M number of output classes. The weighted vote based classifier ensemble selection problem is then stated as follows:

Find the weights of votes V per classifier which will optimize a function $F(V)$. Here, V is an real array of size $N \times M$. $V(i, j)$ denotes the weight of vote of the i^{th} classifier for the j^{th} class. More weight is assigned for that particular class for which the classifier is more confident, whereas the output classes for which the classifier is less confident are given less weight. $V(i, j) \in [0, 1]$ denotes the

² We use 'classifier ensemble' and 'ensemble classifier' interchangeably.

degree of confidence of the i^{th} classifier for the j^{th} class. These weights are used while combining the outputs of the classifiers using weighted voting. Here, F is a classification quality measure of the combined weighted vote based classifier. The particular type of problem like NER has mainly three different kinds of classification quality measures, namely recall, precision and F-measure. Thus, $F \in \{\text{recall, precision, F-measure}\}$.

Multiobjective Formulation. The MOO can be formally stated as follows [7]. Find the vectors $\bar{x}^* = [x_1^*, x_2^*, \dots, x_n^*]^T$ of decision variables that simultaneously optimize the M objective values $\{f_1(\bar{x}), f_2(\bar{x}), \dots, f_M(\bar{x})\}$, while satisfying the constraints, if any.

Now, the weighted vote based classifier ensemble selection problem under the MOO framework takes the form as follows:

Find the weights of votes per classifier V such that, *maximize* $[F_1(V), F_2(V)]$, where $F_1, F_2 \in \{\text{recall, precision, F-measure}\}$ and $F_1 \neq F_2$. We choose $F_1 = \text{recall}$ and $F_2 = \text{precision}$.

Selection of Objectives. Performance of MOO largely depends on the choice of the objective functions which should be as contradictory as possible. In this work, we choose *recall* and *precision* as two objective functions. From the definitions, it is clear that while *recall* tries to increase the number of tagged entries as much as possible, *precision* tries to increase the number of correctly tagged entries. These two capture two different classification qualities. Often, there is an inverse relationship between *recall* and *precision*, where it is possible to increase one at the cost of reducing the other. For example, an information retrieval system (such as a search engine) can often increase its *recall* by retrieving more documents at the cost of increasing number of irrelevant documents retrieved (i.e. decreasing *precision*). This is the underlying motivation of simultaneously optimizing these two objectives. Figure 1 shows, for example, the Pareto optimal front identified by the proposed MOO approach. This again supports the contradictory nature of these two objective functions.

Note, that F-measure is the harmonic mean (i.e., weighted average) of *recall* and *precision*. But, it has been thoroughly discussed in Chapter 2 of Ref [7] that weighted sum approach cannot identify all non-dominated solutions. Only solutions located on the convex part of the Pareto front can be found. But as discussed in the last note of introduction, our another important motivation of this work is to provide the user a set of alternative solutions. Thus, MOO is indeed the best candidate to solve this problem. Here, no weight is required to combine the objectives (i.e., *recall* and *precision*) and thus no *a priori* information on the problem is needed.

Nondominated Sorting GA-II. Our main objective is to find the appropriate weights of voting that will be most suitable to form a classifier ensemble. In order to achieve this goal, we use a multiobjective evolutionary algorithm, namely Nondominated Sorting GA-II (NSGA-II). NSGA-II [12] is a widely used MOO technique based on GA. Here, initially a random parent population P_0

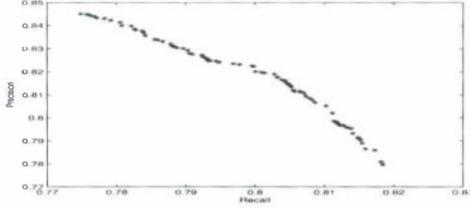


Fig. 1. Pareto optimal front of the proposed MOO based ensemble for the combined classifiers

is created and the population is sorted based on the *partial order* defined by the non-domination relation. This relation yields a sequence of nondominated fronts. Each solution of the population is assigned a fitness which is equal to its non-domination level in the partial order. A child population Q_0 of size N is created from the parent population P_0 by using binary tournament selection, recombination, and mutation operators. According to this algorithm, in the t^{th} iteration, a combined population $R_t = P_t + Q_t$ is formed. The size of R_t is $2N$. All the solutions of R_t are sorted according to non-domination. If the total number of solutions belonging to the best nondominated set F_1 is smaller than N , then F_1 is totally included in $P_{(t+1)}$. The remaining members of the population $P_{(t+1)}$ are chosen from subsequent nondominated fronts in the order of their ranking. To choose exactly N solutions, the solutions of the last included front are sorted using the crowded comparison operator [12] and the best among them (i.e. those with lower crowding distance) are selected to fill in the available slots in $P_{(t+1)}$. The new population $P_{(t+1)}$ is then used for selection, crossover and mutation to create a population $Q_{(t+1)}$ of size N .

3 Named Entity Features

We use the following features for constructing the various classifiers based on the ME framework.

1. **Context words:** These are the preceding and succeeding words of the current word.
2. **Word suffix and prefix:** Fixed length (say, n) word suffixes and prefixes are very effective to identify NEs and work well for the highly inflective Indian language like Bengali. Actually, these are the fixed length character sequences stripped from either the rightmost or leftmost positions of the words.
3. **First word:** This is a binary valued feature that checks whether the current token is the first word of the sentence or not. We consider this feature with the observation that the first word of the sentence is most likely a NE, especially in a newspaper corpus.

4. **Length of the word:** This binary valued feature checks whether the length of the token is less than a predetermined threshold (set to 5) value. We observed that very short words are most probably not the NEs.
5. **Infrequent word:** A list is prepared that contains those words, having less than 10 occurrences in the training data. A binary valued feature 'INFRQ' is defined that fires if the current word appears in this list. We observed that very frequently occurring words are most probably not the NEs.
6. **Part of Speech (POS) information:** POS information of the current and/or the surrounding word(s) are extracted using a SVM based POS tagger [13].
7. **Position of the word:** This binary valued feature checks the position of the word in the sentence. Sometimes, position of the word in a sentence acts as a good indicator for NE identification. This feature fires if the word is at the last position in the sentence.
8. **Digit features:** Several digit features (digitComma, digitPercentage etc.) are defined depending upon the presence and/or the number of digits and/or symbols in a token. These features are helpful to identify miscellaneous NEs.

4 Our Proposed Method for Classifier Ensemble Selection

In this section, we present the classifier ensemble selection problem with a framework that is founded on the principle of MOO algorithm, namely NSGA-II.

4.1 Chromosome Representation and Population Initialization

If the total number of available classifiers is M and total number of output tags (or, classes) is O , then the length of the chromosome is $M \times O$ (each chromosome encodes the weights of votes for possible O classes for each classifier). In the present work, we use real encoding. The entries of each chromosome are randomly initialized to a real value (r) between 0 and 1. Here, $r = \frac{rand()}{RAND_MAX+1}$. As an example, the encoding of a particular chromosome is represented below:

0.59 0.12 0.56 0.09 0.91 0.02 0.76 0.5 0.21

Here, $M = 3$ and $O = 3$ (i.e., total 9 votes can be possible). The chromosome represents the following voting ensemble:

The weights of votes for 3 different output classes are 0.59, 0.12 and 0.56, respectively for classifier 1; 0.09, 0.91 and 0.02, respectively for classifier 2; and 0.76, 0.5 and 0.21, respectively for classifier 3.

If the population size is P then all the P number of chromosomes of this population are initialized in the above way.

4.2 Fitness Computation

Initially, the F-measure values of all the ME based classifiers are computed on a development set. Then, we execute the following steps to compute the objective values.

- 1) Suppose, there are total M number of classifiers. Let, the overall F-measure values of these M classifiers be $F_i, i = 1 \dots M$.
- 2) Each classifier is trained using the training data and tested with the development data. Now, for the ensemble classifier the output label for each word in the development data is determined using the weighted voting of these M classifiers' outputs. The weight of the class provided by the i^{th} classifier is equal to $I(m, i)$. Here, $I(m, i)$ is the entry of the chromosome corresponding to m^{th} classifier and i^{th} class. The combined score of a particular class for a particular word w is:

$$f(c_i) = \sum I(m, i) \times F_m,$$

$$\forall m = 1 : M \ \& \ op(w, m) = c_i$$

Here, $op(w, m)$ denotes the output class provided by the m^{th} classifier for the word w . The class receiving the maximum combined score is selected as the joint decision.

- 3) Now, the overall recall, precision and F-measure values of the ensemble classifier are computed on the development set. The objective functions corresponding to a particular chromosome are $f_1 = \text{recall}$ and $f_2 = \text{precision}$. The main goal is to maximize these two objective functions using the search capability of NSGA-II.

4.3 Genetic Operators

We use crowded binary tournament selection as in NSGA-II, followed by conventional crossover and mutation. The most characteristic part of NSGA-II is its elitism operation, where the non-dominated solutions [7] among the parent and child populations are propagated to the next generation. The near-Pareto-optimal strings of the last generation provide the different solutions to the ensemble problem.

4.4 Selection of a Solution from the Final Pareto Optimal Front

In MOO, the algorithms produce a large number of non-dominated solutions [7] on the final Pareto optimal front. Each of these solutions provides a weighted vote based classifier ensemble. All the solutions are equally important from the algorithmic point of view. But, sometimes the user may need only a single solution. Consequently, in this paper a method of selecting a single solution from the set of solutions is now developed.

For every solution on the final Pareto optimal front, the F-measure value of the weighted vote based classifier ensemble for the development set is calculated. The best solution is selected to be the one, having the highest F-measure value. Final results on the test data are reported using the classifier ensemble corresponding to this best solution. There can be many other different approaches of selecting a solution from the final Pareto optimal front.

5 Experimental Results and Discussions

We use the OpenNLP Java based ME package³ for the MaxEnt experiments. Model parameters are computed with 200 iterations without any feature frequency cutoff. We set the following parameter values for NSGA-II: population size=100, number of generations=50, probability of mutation=0.2 and probability of crossover = 0.9. Note that these values are selected after a thorough sensitivity analysis of the parameter values on the performance of the proposed system. We define three different *baseline* ensembles as below:

1. *Baseline 1*: This is based on the majority voting among the classifiers.
2. *Baseline 2*: This is a weighted voting approach. In each classifier, weights are calculated based on the average F-measure value of the 3-fold cross validation on the training data.
3. *Baseline 3*: For each classifier, the average F-measure value of each class is computed from the 3-fold cross validation on the training data. The weight of any classifier is set to the average F-measure value of the corresponding class that it assigns to a word.

5.1 Datasets for NER

Indian languages are resource-constrained in nature. For NER, we use a Bengali news corpus [8], developed from the archive of a leading Bengali newspaper available in the web. Out of 34 million wordforms, a portion containing approximately 250K wordforms is manually annotated with a coarse-grained NE tagset of four tags namely, PER (*Person name*), LOC (*Location name*), ORG (*Organization name*) and MISC (*Miscellaneous name*). The miscellaneous name includes date, time, number, percentages, monetary and measurement expressions. The data is collected mostly from the *National, States, Sports* domains and the various sub-domains of *District* of the particular newspaper. This annotation was carried out by one of the authors and verified by an expert. We also use the IJCNLP-08 NER on South and South East Asian Languages (NERSSEAL)⁴ Shared Task data of around 100K wordforms that were originally annotated with a fine-grained tagset of twelve tags. This data is mostly from the *agriculture* and *scientific* domains. An appropriate mapping is defined to convert the fine-grained NE annotated data to the desired forms, i.e. tagged with a coarse-grained tagset of four tags. In order to report the evaluation results, we randomly partition the dataset into training, development and test sets that contain approximately 263K, 50K and 37K wordforms, respectively. The number of unseen NEs in the test set is 39.5%. In order to properly denote the boundaries of NEs, four basic NE tags are further divided into the format I-TYPE (TYPE→PER/LOC/ORG/MISC) which means that the word is inside a NE of type TYPE. Only if two NEs of the same type immediately follow each other, the first word of the second NE will have tag B-TYPE to show that it starts a new NE. This is the standard IOB format that was followed in the CoNLL-2003 shared task [10].

³ <http://maxent.sourceforge.net/>

⁴ <http://ltrc.iit.ac.in/ner-ssea-08>

5.2 Results and Discussions

We build a number of different ME models by considering the various combinations of the available NE features. In particular, we construct the classifiers from the following set of features:

(i). considering the various context size within the previous three and next three words, (ii). word suffixes and prefixes of length up to three (3+3 different features) or four (4+4 different features) characters, (iii). POS information of the current word, (iv). first word, (v). length, (vi). infrequent word, (vii). position, and (viii). digit features.

We generate 152 different classifiers by considering the various combinations of the available features. Some of these classifiers are shown in Table 1. The best

Table 1. Evaluation results with the various feature subsets. Here, the following abbreviations are used: CW:Context words, PS: Size of the prefix, SS: Size of the suffix, WL: Word length, IW: Infrequent word, PW: Position of the word, FW:First word, DI: Digit information, -i,j: Context words spanning from the i^{th} left position to the j^{th} right position with the current word at position 0, R: recall, P: precision, F:F-measure, X: Denotes the presence of the corresponding feature (we report percentages).

Classifier	CW	FW	PS	SS	WL	IW	PW	DI	POS	Normal Training			After Sampling		
										R	P	F	R	P	F
M_{34}	-2,1	X	3	3	X	X	-	X	X	71.21	83.55	76.88	81.52	69.79	75.20
M_{42}	-2,1	X	3	3	X	X	X	X	X	70.87	83.73	76.74	81.52	69.85	75.24
M_{82}	-2,1	X	3	4	X	-	-	X	X	68.65	83.54	75.36	79.41	69.38	74.06
M_{83}	-2,0	X	3	4	X	-	-	X	X	67.54	82.20	74.15	80.18	69.43	74.42
M_{85}	-1,2	X	3	4	X	-	-	X	X	68.01	82.21	74.44	80.09	68.82	74.03
M_{89}	-2,2	X	4	3	X	-	-	X	X	65.32	81.89	72.67	78.75	67.77	72.85
M_{90}	-2,1	X	4	3	X	-	-	X	X	66.24	82.27	73.39	79.21	68.75	73.61
M_{92}	-1,1	X	4	3	X	-	-	X	X	68.85	82.84	75.20	78.03	67.35	72.30
M_{93}	-1,2	X	4	3	X	-	-	X	X	66.61	81.67	73.37	79.91	69.06	74.09
M_{105}	-2,2	X	3	4	X	X	-	X	X	67.40	82.67	74.26	73.79	63.02	67.98
M_{106}	-2,1	X	3	4	X	X	-	X	X	69.10	83.45	75.60	79.91	68.84	73.96
M_{108}	-1,1	X	3	4	X	X	-	X	X	69.42	82.63	75.45	79.25	67.16	72.70
M_{109}	-1,2	X	3	4	X	X	-	X	X	68.28	81.95	74.50	80.45	68.48	73.98
M_{110}	0,2	X	3	4	X	X	-	X	X	67.88	81.18	73.94	80.05	68.75	73.97
M_{112}	3,3	X	3	4	X	X	-	X	X	65.50	81.20	72.51	80.09	65.43	72.02
M_{113}	-2,2	X	4	3	X	X	-	X	X	65.93	81.76	72.99	79.34	67.26	72.80
M_{114}	-2,1	X	4	3	X	X	-	X	X	66.72	82.30	73.70	79.48	68.32	73.48
M_{116}	-1,1	X	4	3	X	X	-	X	X	69.15	82.71	75.32	78.21	67.30	72.34
M_{116}	-1,2	X	4	3	X	X	-	X	X	66.79	81.68	73.49	79.91	68.48	73.76
M_{129}	-2,2	X	3	4	X	X	X	X	X	67.36	83.07	74.39	74.67	62.48	68.03
M_{130}	-2,1	X	3	4	X	X	X	X	X	68.85	83.54	75.49	79.98	68.91	74.03
M_{133}	-1,2	X	3	4	X	X	X	X	X	68.15	82.16	74.50	80.50	68.52	74.03
M_{137}	-2,2	X	4	3	X	X	X	X	X	65.68	82.01	72.94	79.25	67.23	72.75
M_{140}	-1,1	X	4	3	X	X	X	X	X	68.94	82.84	75.25	78.16	67.31	72.33
M_{141}	-1,2	X	4	3	X	X	X	X	X	66.54	81.90	73.42	79.84	68.51	73.75

Table 2. Results on the test set. Here R, P and F refer to recall, precision and F-measure, respectively (we report percentages)

Model	Normal Training			After Sampling			Mixed		
	R	P	F	R	P	F	R	P	F
Best individual classifier	71.21	83.54	76.88	81.52	69.85	75.24	71.21	83.54	76.88
<i>Baseline 1</i>	71.25	84.12	77.15	81.90	70.21	75.61	71.50	83.98	77.24
<i>Baseline 2</i>	71.34	84.21	77.24	82.06	70.71	75.96	71.69	84.21	77.45
<i>Baseline 3</i>	71.43	85.01	77.63	82.54	71.81	76.80	73.12	84.25	78.29
GA based ensemble	71.68	86.07	78.22	83.19	74.25	78.47	78.35	81.38	79.89
MOO based ensemble	74.00	84.82	79.04	82.14	76.39	79.16	79.98	82.24	81.10

individual classifier shows the recall, precision and F-measure values of 71.21%, 83.54% and 76.88%, respectively. Thereafter, we apply the single objective GA based approach [6] to determine the appropriate classifier ensemble. Overall evaluation results of this ensemble along with the best individual classifier and three different *baseline* ensembles are reported in Table 2. Results show that the single objective GA based ensemble performs better than the best individual classifier as well as the three *baseline* ensembles. Then, we apply our proposed MOO based approach to determine the appropriate ensemble and its results are also shown in Table 2. We observe the increments of 2.16%, 1.89%, 1.80% and 1.41% F-measure values over the best individual classifier, *Baseline 1*, *Baseline 2*, and *Baseline 3*, respectively. The proposed MOO based approach also attains an improvement of 0.82% F-measure over the corresponding single objective version. Statistical analysis of variance, (ANOVA) [14], is performed in order to examine whether the MOO based ensemble technique really outperforms the best individual classifier, three *baseline* ensembles and GA based ensemble. ANOVA tests show that the differences in mean recall, precision and F-measure are statistically significant as p value is less than 0.05 in each of the cases.

Our training set is highly imbalanced. The ratio between positive (NEs) and negative examples is 1:11.21. We observed on the development set that this skewed distribution heavily biases the classifiers towards the negative category, and accordingly investigated random sampling techniques to make the ratio of positive and negative examples more balanced. We experiment with a sampling strategy that randomly over-samples the positive examples until it becomes equal to the number of negative ones. This random sampling yields a new set of 152 classifiers, which are again evaluated on the development set. Results reveal that in most of the cases, recall values are increased at the cost of precisions with respect to their corresponding older versions (constructed with the same set of features). However, the overall F-measure values are quite similar in most of the classifiers. Results of some classifiers on the sampled dataset are reported in Table 1 for the test set. Overall results are presented in Table 2 which shows that the proposed multiobjective based approach performs better than the best individual classifier, three *baseline* ensembles and the single objective GA based ensemble [6]. Comparison between these two sets of results also shows that the

later gains recall at the cost of precision in most of the cases. As a result of sampling, single and multiobjective optimization based techniques attain overall performance improvements by 0.25% and 0.12% F-measure points, respectively.

The basic principle of MOO is that objectives should be as much conflicting as possible in nature. In the first set (normal classifiers), the recall values are lower than the precision values. But in the second set (sampled classifiers), recalls are higher than precisions in general. These two observations give an insight that the capabilities of MOO could be best utilized if it is executed on the combination of these two types of classifiers. Thus, we select 76 best classifiers according to their F-measure values from each of these sets. Thereafter, GA and MOO based approaches are executed on these resultant 152 classifiers. Evaluation results are reported in Table 2 for the test set, which again shows that MOO based approach performs the best. The proposed MOO based ensemble technique performs superior to the previous two MOO based ensembles with more than 2.06% (before sampling) and 1.94% (after sampling) F-measures, respectively. The single objective GA based ensemble also gains 1.67% and 1.42% F-measures, respectively. Compared to the *baseline* models, we observe a slight degradation of precision in the proposed MOO based ensemble. However, the Pareto optimal front of Figure 1 reveals that there indeed exists some solutions with higher precision values.

Summary of Results. Evaluation results reveal that the proposed approach is truly able to improve the performance of the classifiers by appropriately ensembling them. Performance of the ensembles can further be improved if we combine the individual classifiers, having a variety of classification methodologies that could achieve different rate of correctly classified individuals. Moreover, MOO based approach provides a set of trade-off solutions from which users can choose the desired one based on their requirement. We also observe that MOO performs superior to the best individual classifier, *baseline* models and a single objective GA based approach [6].

6 Conclusion

In this paper, we present the problem of selecting the appropriate votes per classifier for each class in NER as an optimization problem. We have assumed and experimentally verified that instead of eliminating some classifiers completely, it is better to quantify the amount of vote per classifier for each class. To solve this problem, we proposed a MOO based solution that can simultaneously optimize two different classification measures. Based on the ME framework, a number of different classifiers have been built by selecting different feature combinations from a set of language independent features. Our proposed algorithm is applicable for any language due to its language independent nature. The proposed algorithm has been evaluated for a resource constrained language like Bengali. Evaluation results showed that the overall performance attained by the proposed technique outperforms the best individual classifier, three different *baseline* ensembles and a single objective optimization based ensemble technique. In future

we would like to develop the vote based classifier ensembles using other learning algorithms like Conditional Random Field and Support Vector Machine.

References

1. Cunningham, H.: GATE, a General Architecture for Text Engineering. *Computers and the Humanities* 36, 223–254 (2002)
2. Babych, B., Hartley, A.: Improving Machine Translation Quality with Automatic Named Entity Recognition. In: *Proceedings of EAMT/EACL 2003 Workshop on MT and other Language Technology Tools*, pp. 1–8 (2003)
3. Moldovan, D., Harabagiu, S., Girju, R., Morarescu, P., Lacatusu, F., Novischi, A., Badulescu, A., Bolohan, O.: LCC Tools for Question Answering. In: *Text REtrieval Conference, TREC 2002* (2002)
4. Nobata, C., Sekine, S., Isahara, H., Grishman, R.: Summarization System Integrated with Named Entity Tagging and IE Pattern Discovery. In: *Proceedings of Third International Conference on Language Resources and Evaluation (LREC 2002)*, Spain (2002)
5. Goldberg, D.E.: *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, New York (1989)
6. Ekbal, A., Saha, S.: Weighted Vote Based Classifier Ensemble Selection Using Genetic Algorithm for Named Entity Recognition. In: *Proceedings of 15th International Conference on Applications of Natural Language to Information Systems (NLDB 2010)*, Cardiff, Wales, UK, pp. 256–267 (2010)
7. Deb, K.: *Multi-objective Optimization Using Evolutionary Algorithms*. John Wiley and Sons, Ltd., England (2001)
8. Ekbal, A., Bandyopadhyay, S.: A Web-based Bengali News Corpus for Named Entity Recognition. *Language Resources and Evaluation Journal* 42(2), 173–182 (2008)
9. Florian, R., Ittycheriah, A., Jing, H., Zhang, T.: Named Entity Recognition through Classifier Combination. In: *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003* (2003)
10. Tjong Kim Sang, E.F., De Meulder, F.: Introduction to the CoNLL-2003 Shared Task: Language Independent Named Entity Recognition. In: *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pp. 142–147 (2003)
11. Ekbal, A., Bandyopadhyay, S.: Voted NER System using Appropriate Unlabeled Data. In: *Proceedings of the 2009 Named Entities Workshop: Shared Task on Transliteration (NEWS 2009), ACL-IJCNLP 2009*, pp. 202–210 (2009)
12. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation* 6(2), 181–197 (2002)
13. Ekbal, A., Bandyopadhyay, S.: Web-based Bengali News Corpus for Lexicon Development and POS Tagging. *POLIBITS* 37, 20–29 (2008)
14. Anderson, T.W., Scolve, S.: *Introduction to the Statistical Analysis of Data*. Houghton Mifflin (1978)

Local Search for Stable Marriage Problems with Ties and Incomplete Lists

Mirco Gelain¹, Maria Silvia Pini¹, Francesca Rossi¹,
Kristen Brent Venable¹, and Toby Walsh²

¹ Dipartimento di Matematica Pura ed Applicata, Università di Padova, Italy
{mgelain,mpini,frossi,kvenable}@math.unipd.it
² NICTA and UNSW Sydney, Australia
Toby.Walsh@nicta.com.au

Abstract. The stable marriage problem has a wide variety of practical applications, ranging from matching resident doctors to hospitals, to matching students to schools, or more generally to any two-sided market. We consider a useful variation of the stable marriage problem, where the men and women express their preferences using a preference list with ties over a subset of the members of the other sex. Matchings are permitted only with people who appear in these preference lists. In this setting, we study the problem of finding a stable matching that marries as many people as possible. Stability is an envy-free notion: no man and woman who are not married to each other would both prefer each other to their partners or to being single. This problem is NP-hard. We tackle this problem using local search, exploiting properties of the problem to reduce the size of the neighborhood and to make local moves efficiently. Experimental results show that this approach is able to solve large problems, quickly returning stable matchings of large and often optimal size.

1 Introduction

The stable marriage problem [1] is a well-known problem of matching men to women to achieve a certain type of “stability”. Each person expresses a strict preference ordering over the members of the opposite sex. The goal is to match men to women so that there are no two people of opposite sex who would both rather be matched with each other than with their current partners. Surprisingly such a stable marriage always exists and one can be found in polynomial time. Gale and Shapley give a quadratic time algorithm to solve this problem based on a series of proposals of the men to the women (or vice versa) [2]. The stable marriage problem has a wide variety of practical applications, ranging from matching resident doctors to hospitals, sailors to ships, primary school students to secondary schools, as well as in market trading.

There are many variants of this classical formulation of the stable marriage problem. Some of the most useful in practice include incomplete preference lists (SMI), that allows us to model unacceptability for certain members of the other

sex, and preference lists with ties (SMT), that model indifference in the preference ordering. With a SMI problem, we have to find a stable marriage in which the married people accept each other. It is known that all solutions of a SMI problem have the same size [3] (that is, number of married people). In SMT problems, instead, solutions are stable marriages where everybody is married. Both of these variants are polynomial to solve. In real world situations, both ties and incomplete preference lists may be needed. Unfortunately, when we allow both, the problem becomes NP-hard [3]. In a SMTI (Stable Marriage with Ties and Incomplete lists) problem, there may be several stable marriages of different sizes, and solving the problem means finding a stable marriage of maximum size.

In this paper we investigate the use of a local search approach to tackle this problem. Our algorithm starts from a randomly chosen marriage and, at each step, moves to a neighbor marriage which is obtained by removing one blocking pair, that is, a man-woman pair who are not married to each other in the current marriage but who prefer to be married with each other rather than with their current partners. Stable marriages have no blocking pairs, so the aim of such a move is to pass to a marriage which is closer to stability. Among the neighbor marriages, the evaluation function chooses one with the smallest number of blocking pairs and of singles. Since there may be several stable marriages with different sizes, we look for the one with maximum size (that is, the smallest number of singles). Random moves are also used, to avoid stagnation in local minima. The algorithm stops when a perfect matching (that is, a stable marriage with no singles) is found, or when a given limit on the number of steps is reached.

This basic local search approach works well with problems of limited size, but does not scale. With large sizes, it fails to find good solutions and sometimes even stable marriages. One of the main reasons is that the neighborhood can be very large, since a marriage may have a large number of blocking pairs. Many such blocking pairs can be ignored since they are "dominated" by others, whose removal will also eliminate all the dominated blocking pairs. By considering only undominated blocking pairs, we can solve SMTI problems of much larger size in a small amount of time. The marriages returned by our local search method are stable and contain very few single people. Experiments on randomly generated SMTI problems of size 100 show that our algorithm is able to find stable marriages with at most two singles on average in tens of seconds at worst.

The SMTI problem has been tackled also in [4], where the problem is modeled as a constraint optimization problem and a constraint solver is employed to solve it. This systematic approach is guaranteed to find always an optimal solution. However, our experimental results show that our local search algorithm in practice always finds optimal solutions. Moreover, it scales well to sizes much larger than those considered in [4]. Instances of size comparable to ours are considered in [5]. However, the problem solved in that paper is the decision version of our optimization problem. That is, they ask if there exists a stable marriage of a certain size. Another approach is to use approximation. Given an SMTI problem, if its maximum cardinality stable marriage marriages are of size k , an

α/β -approximation algorithm is able to return a stable marriage of size at least $\beta/\alpha \cdot k$. The SMTI problem cannot have an i -approximation algorithm for i greater than $33/29$ unless $P=NP$ [6]. A $3/2$ -approximation algorithm has been proposed in [7].

2 Background

2.1 Stable Marriage Problems with Ties and Incompleteness

A stable marriage (SM) problem [1] consists of matching members of two different sets, usually called men and women. When there are n men and n women, the SM problem is said to have size n . Each person strictly ranks all members of the opposite sex. The goal is to match the men with the women so that there are no two people of opposite sex who would both rather marry each other than their current partners. Such a marriage is called *stable*. At least one stable marriage exists for every SM problem. In fact, the set of stable marriages forms a lattice. Gale and Shapley give a polynomial time algorithm to find the stable marriage at the top (or bottom) of this lattice [2].

In this paper we consider a variant of the SM problem where preference lists may include ties and may be incomplete. This variant is denoted by SMTI [8]. Ties express indifference in the preference ordering, while incompleteness models unacceptability for certain partners.

Definition 1 (SMTI marriage). *Given a SMTI problem with n men and n women, a marriage M is a one-to-one matching between men and women such that partners are acceptable for each other. If a man m and a woman w are matched in M , we write $M(m) = w$ and $M(w) = m$. If a person p is not matched in M we say that he/she is single.*

Definition 2 (Marriage size). *Given a SMTI problem of size n and a marriage M , its size is the number of men (or women) that are married.*

An example of a SMTI problem with four men and women is shown in Table 1. A SMTI problem is described by giving, for each man and woman, the corresponding preference list over members of the other sex. For example, by writing $2 : 2 (3\ 4)$ among the men's preference lists we mean that man m_2 strictly prefers woman w_2 to women w_3 and w_4 , that are equally preferred.

Table 1. An example of a SMTI problem of size 4

men's preference lists	women's preference lists
1: 2 1	1: 3 1 (2 4)
2: 2 (3 4)	2: 1 4 2
3: (1 2 3 4)	3: (1 2) (4 3)
4: (3 2) 1 4	4: (3 2 4)

Definition 3 (Blocking pairs in SMTIs). Consider a SMTI problem P , a marriage M for P , a man m and a woman w . A pair (m, w) is a blocking pair in M if m and w find acceptable each other and m is either single in M or he strictly prefers w to $M(m)$, and w is either single in M or she strictly prefers m to $M(w)$.

Definition 4 (Weakly Stable Marriages). Given a SMTI problem P , a marriage M for P is weakly stable if it has no blocking pairs.

As we will consider only weakly stable marriages, we will simply call them stable marriages. Given a SMTI problem, there may be several stable marriages of different size. If the size of a marriage coincides with the size of the problem, it is said to be a perfect matching.

In the above example, the marriage 2 3 1 4 (where the number in position i indicates the woman married to man m_i in that marriage) is stable and its size is 4, so it is a perfect matching.

Solving a SMTI problem means finding a stable marriage with maximal size. This problem is NP-hard [3].

2.2 Local Search

Local search [9,10] is one of the fundamental paradigms for solving computationally hard combinatorial problems. Local search methods in many cases represent the only feasible way for solving large and complex instances. Moreover, they can naturally be used to solve optimization problems.

Given a problem instance, the basic idea underlying local search is to start from an initial search position in the space of all solutions (typically a randomly or heuristically generated candidate solution, which may be infeasible, sub-optimal or incomplete), and to improve iteratively this candidate solution by means of typically minor modifications. At each *search step* we move to a position selected from a *local neighborhood*, chosen via a heuristic evaluation function. The evaluation function typically maps the current candidate solution to a real number and it is such that its global minima correspond to solutions of the given problem instance. The algorithm moves to the neighbor with the smallest value of the evaluation function.

This process is iterated until a *termination criterion* is satisfied. The termination criterion is usually the fact that a solution is found or that a predetermined number of steps is reached, although other variants may stop the search after a predefined amount of time.

Different local search methods vary in the definition of the neighborhood and of the evaluation function, as well as in the way in which situations are handled when no improvement is possible. To ensure that the search process does not stagnate in unsatisfactory candidate solutions, most local search methods use randomization: at every step, with a certain probability a random move is performed rather than the usual move to the best neighbor.

3 Local Search on SMTIs

We adapt the classical local search schema to SMTI problems as follows. Given a SMTI problem P , we start from a randomly generated marriage M for P . At each search step, we move to a new marriage in the neighborhood of the current one. For each marriage M , the neighborhood $N(M)$ is the set of all marriages obtained by removing one blocking pair from M . Consider a blocking pair $bp = (m, w)$ in M and assume $m' = M(w)$ and $w' = M(m)$. Then, removing bp from M means obtaining a marriage M' in which m is married with w and both m' and w' become single, leaving the other pairs in the marriage M unchanged. Notice that, if M is stable, its neighborhood is empty. Notice also that this notion of neighborhood is not symmetric.

To select the neighbor to move to, we use an evaluation function $f : \mathcal{M}_n \rightarrow \mathbb{Z}$, where \mathcal{M}_n is the set of all possible marriages of size n , and $f(M) = nbp(M) + ns(M)$. For each marriage M , $nbp(M)$ is the number of blocking pairs in M , while $ns(M)$ is the number of singles in M which are not in any blocking pair. The algorithm moves to a marriage $M' \in N(M)$ such that $f(M') \leq f(M'')$ $\forall M'' \in N(M)$.

During the search, the algorithm maintains the best marriage found so far, defined as follows: if no stable marriage has been found, then the best marriage is the one with the smallest value of the evaluation function; otherwise, it is the stable marriage with less singles.

To avoid stagnation in a local minimum of the evaluation function, at each search step we perform a random walk with probability p (where p is a parameter of the algorithm). In the random walk, we move to a randomly selected marriage in the neighborhood (we tried also to move to a generic random marriage, but this gave worse behavior). If a stable marriage is reached, its neighborhood is empty and a random restart is performed.

The algorithm terminates if a perfect marriage (that is, a stable marriage with no singles) is found, or when a maximal number of search steps is reached. Upon termination, the algorithm returns the best marriage found during the search.

The pseudo-code of our algorithm, called LTI, is shown in Algorithm 1. In the pseudo-code, M_{best} is the best marriage found so far, and f_{best} its evaluation (number of blocking pairs plus number of singles). Function *best_neighbor* returns one of the best marriages in the neighborhood of the current marriage, according to the evaluation function.

In addition to this simple local search algorithm which directly applies standard local search approaches to SMTI problems, we have also designed a more sophisticated algorithm which has been tailored to exploit the specific features of SMTI problems. The main difference is in the definition of the neighborhood, which refers to the notion of *undominated* blocking pairs.

Definition 5 (Dominance in blocking pairs). *Let (m, w) and (m, w') be two blocking pairs. Then (m, w) dominates (from the men's point of view) (m, w') if m prefers w to w' . There is an equivalent concept from the women's point of view.*

Definition 6 (Undominated blocking pair). A men- (resp., women-) undominated blocking pair is a blocking pair such that there is no other blocking pair that dominates it from the men's (resp., women's) point of view. When the point of view (men or women) is clear or not important, we will omit it.

Algorithm 1. LTI

input : a SMTI problem P , an integer max_steps , a probability p
output: a marriage

```

1  $M \leftarrow$  random marriage
2  $steps \leftarrow 0$ 
3  $M_{best} \leftarrow M$ 
4  $f_{best} \leftarrow f(M)$ 
5 repeat
6   if  $f(M) = 0$  then
7      $\lfloor$  return  $M$ 
8   if  $rand() \leq p$  then
9      $\lfloor M \leftarrow RandomWatk(M)$ 
10  else
11     $PAIRS \leftarrow$  blocking pairs in  $M$ 
12    if  $PAIRS$  is empty then
13       $\lfloor$  perform a random restart
14    else
15       $\lfloor M \leftarrow best\_neighbor(M, PAIRS)$ 
16  if  $M$  is the first stable marriage found so far then
17     $\lfloor f_{best} \leftarrow f(M), M_{best} \leftarrow M$ 
18  if  $M_{best}$  is not stable and  $f_{best} > f(M)$  then
19     $\lfloor f_{best} \leftarrow f(M), M_{best} \leftarrow M$ 
20  if both  $M_{best}$  and  $M$  are stable and  $f_{best} > f(M)$  then
21     $\lfloor f_{best} \leftarrow f(M), M_{best} \leftarrow M$ 
22   $steps \leftarrow steps + 1$ 
23 until  $steps > max\_steps$  ;
24 return  $M_{best}$ 

```

For example, consider the SMTI problem in Table 1, the marriage 1 2 3 4, and two blocking pairs (m_1, w_2) and (m_4, w_2) . Using the definitions above, (m_1, w_2) dominates (m_4, w_2) from the women's point of view. If we remove (m_4, w_2) from the marriage, (m_1, w_2) will remain. On the other hand, removing (m_1, w_2) also eliminates (m_4, w_2) . Thus removing undominated blocking pairs may reduce the number of blocking pairs more than eliminating dominated pairs.

We call LTIU the algorithm LTI where the neighborhood is defined as the set of marriages obtained from the current one by removing any dominated blocking pair. More precisely, at each step we consider the undominated blocking pairs from the men's point of view which are also undominated from women's point

of view. Notice that, in this step, the role of men and women matters, and will yield a different result if swapped.

Then, to ensure gender neutrality in our algorithm¹, in the next step we swap genders and do the same.

Due to their ability to restart, our algorithms have the PAC (probabilistically approximate complete property) [11]. That is, as their runtime goes to infinity, the probability that the algorithm returns an optimal solution goes to one. If the algorithm starts at a stable marriage, the algorithms will perform a random restart, which will end up in an optimal solution with probability greater than zero. On the other hand, if the algorithm starts from a non-stable marriage, we perform one or more steps in which we remove a blocking pair. This sequences of blocking pair removal have been shown to converge to a stable marriage with non-zero probability in the context of SMs with incomplete preference lists [12]. The proof of this result can be adapted to our context, as we have ties in the preference lists. Since a stable marriage can be reached with non-zero probability, and as we have argued above that from any stable marriage random restarting will reach an optimal solution with non-zero probability, the PAC property holds.

4 Experimental Setting

Problems are generated using the same method as in [4]. The generator takes three parameters: the problem's size n , the probability of incompleteness p_1 and the probability of ties p_2 . Given a triple (n, p_1, p_2) , a SMTI problem with n men and n women is generated, as follows:

1. For each man and woman, we generate a random preference list of size n , i.e., a permutation of n persons;
2. We then iterate over each man's preference list: for a man m_i and for each woman w_j in his preference list, with probability p_1 we delete w_j from m_i 's preference list and delete m_i from w_j 's preference list. In this way we get a possibly incomplete preference list.
3. If any man or woman has an empty preference list, we discard the problem and go to step 1.
4. We iterate over each person's (men and women's) preference list as follows: for a man m_i and for each woman in his preference list, in position $j \geq 2$, with probability p_2 we set the preference for that woman as the preference for the woman in position $j - 1$ (thus putting the two women in a tie).

Note that this method generates SMTI problems in which the acceptance is symmetric. In fact, if a woman w is not acceptable for a man m , m is removed from w 's preference list. This does not introduce any loss of generality because, even if such a removal is not performed, m and w cannot be matched together in any stable marriage.

¹ Gender neutrality is usually considered a desirable feature in a stable marriage procedure.

Notice also that this generator will not construct a SMTI problem in which a man (resp., woman) has in his preference list only women (resp., men) who do not find him (resp. her) acceptable. Such a man (resp., woman) will remain single in every stable matching. Therefore a simple preprocessing step can remove such men and women, giving a smaller problem of the form constructed by our generator.

We generated random SMTI problems of size 100, by letting p_2 vary in $[0, 1.0]$ with step 0.1, and p_1 vary in $[0.1, 0.8]$ with step 0.1 (above 0.8 the preference lists start to be empty). For each parameter combination, we generated 100 problem instances. Moreover, the probability of the random walk is set to $p=20\%$ and the search step limit is $s=50000$.

4.1 Experimental Results

We run our experiments on 2 x Quad-Core AMD Opteron 2.3GHz CPU with 2GB of RAM. In practice we used only one core because our algorithm is not designed for multi threading.

We first analyzed the behavior of the base algorithm, LTI. Unfortunately this algorithm fails to find a stable marriage in most of our test problems (see Figure 1). In fact, LTI always finds a stable marriage where there are many ties (that is, p_2 high) and/or a lot of incompleteness (that is, p_1 high).

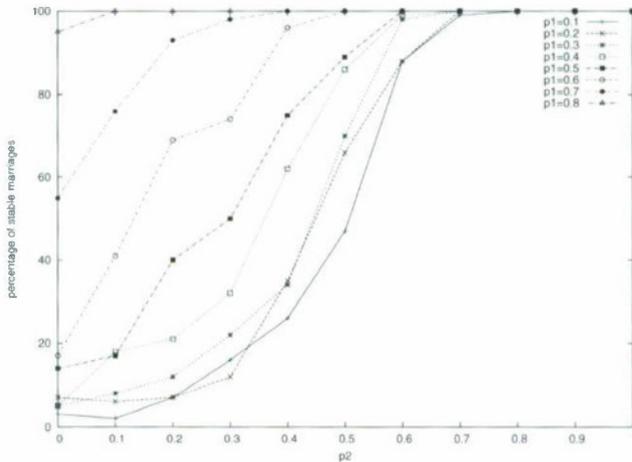


Fig. 1. Average number of stable marriages found by LTI

On the other hand, algorithm LTIU finds a stable marriage in 100% of the runs. Since stability is essential in our context, from now on we will only show the experimental results for algorithm LTIU.

We start by showing the average size of the marriages returned by LTIU. In Figure 2 we can see that LTIU finds a perfect marriage (that is, a stable

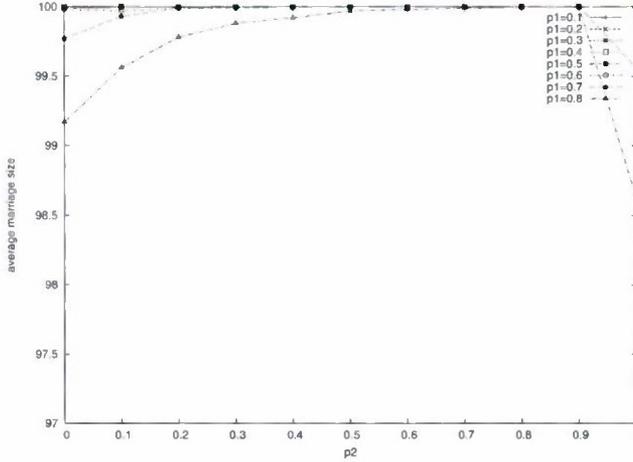


Fig. 2. Average size of marriages with LTIU

marriage with no singles) almost always. Even in settings with a large amount of incompleteness (that is, $p_1 = 0.7 - 0.8$) the algorithm finds very large marriages, with only 2 singles on average.

We also consider the number of steps needed by our algorithm. From Figure 3(a), we can see that the number of steps is less than 2000 most of the time, except for problems with a large amount of incompleteness (i.e. $p_1 = 0.8$). As expected, with p_1 greater than 0.6, the algorithm requires more steps. In some cases, it reaches the step limit of 50000. Moreover, as the percentage of ties rises, stability becomes easier to achieve and thus the number of steps tends to decrease slightly. We note that complete indifference (i.e. $p_2=1$) is a special case. In fact, in this situation, the number of steps increases for almost every value of p_1 . This is because the algorithm makes most of its progress via random restarts. In these problems every person in a preference list is equally preferred to all the others. This means that the only blocking pairs are those involving singles who both find acceptable each other. In this situation, after a few steps all singles that can be married are matched, stability is reached, and the neighborhood becomes empty. The algorithm therefore performs another random restart. It is therefore very difficult to reach a perfect matching and the algorithm often runs until the step limit.

The algorithm takes, on average, less than 40 seconds to give a result even for problems with a lot of incompleteness (see Figure 3(b)). As expected, with $p_2 = 1$ the time increases for the same reason discussed above concerning the number of steps.

Re-considering Figure 2 and the fact that all the marriages the algorithm finds are stable, we notice that most of the marriages are perfect.

From Figure 4 we see that the average percentage of matchings that are perfect is almost always 100% and this percentage only decreases when the incompleteness is large.

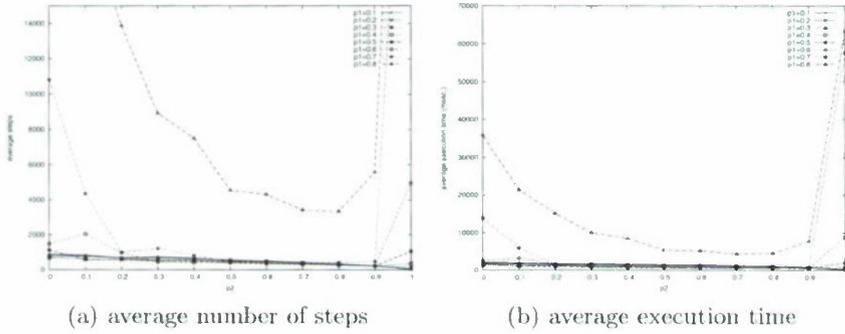


Fig. 3. Average number of steps and execution time for LTIU

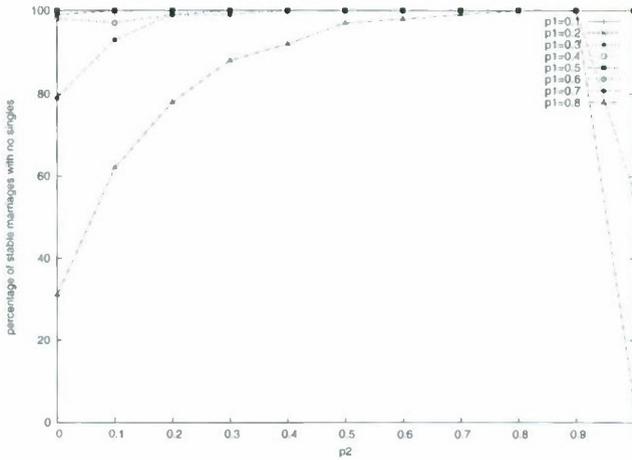


Fig. 4. Percentage of perfect matchings

We compared our local search approach to the complete method from [4]. In their experiments, they measured the maximum size of the stable marriages in problems of size 10, fixing p_1 to 0.5 and varying p_2 in $[0,1]$. We did the same experiments (generating new instances), and obtained stable marriages of a very similar size to those reported in [4]. This means that although our algorithm is incomplete in principle, it always finds an optimal solution in our randomly generated instances, and for small sizes it behaves as a complete algorithm in terms of size of the returned marriage. However, we can also tackle problems of much larger sizes (at least 100), still obtaining optimal solutions most of the times.

We also considered the runtime behavior of our algorithm. In Figure 5 we show the average normalized number of blocking pairs and, in Figure 6, the average normalized number singles of the best marriage as the execution proceeds. Although the step limit is 50000, we only plot results for the first steps because the

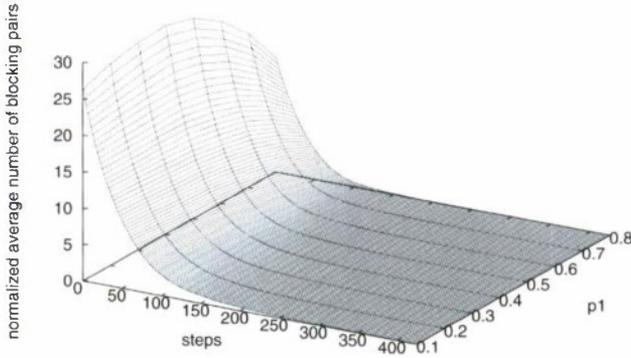


Fig. 5. Average normalized number of blocking pairs ($p_2=0.5$)

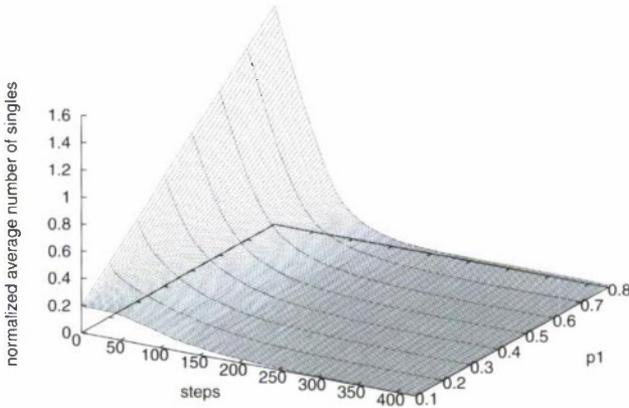


Fig. 6. Average normalized number of singles ($p_2=0.5$)

rest is a long plateau that is not very interesting. We shows the results only for $p_2 = 0.5$. However, for greater (resp., lower) number of ties the curves are shifted slightly down (resp., up). From Figure 5 we can see that the average number of blocking pairs decreases very fast, reaching 5 blocking pairs after only 100 steps. Then, after 300-400 steps, we reach 0 blocking pairs (i.e. a stable marriage) almost all the times for all values of p_1 . Considering Figure 6, we can see that the algorithm starts with more singles for greater values of p_1 . This happens because, with more incompleteness, it is more unprobable for a person to be acceptable. However, after 200 steps, the average number of singles becomes very small no matter the incompleteness in the problem. Looking at both Figures 5 and 6, we observe that, although we set a step limit $s = 50000$, the algorithm reaches a very good solution after just 300-400 steps. In fact, after this number of steps, the best marriage found by the algorithm usually has no blocking pairs nor singles, i.e. it is a perfect matching. This appears largely independent of the

amount of incompleteness and the number of ties in the problems. Hence, for SMTI problems of size 100 we could set the step limit to just 400 steps and still be reasonably sure that the algorithm will return a stable marriage with a large size, no matter the amount of incompleteness and ties.

5 Conclusions

We have presented a local search approach for solving stable marriage problems with ties and indifference. Experimental results show that our algorithm is both fast and effective at finding large stable marriages. Moreover, the runtime behavior of the algorithms is not greatly influenced by the amount of incompleteness or ties in the problem. The algorithm was usually able to obtain a very good solution after a very small amount of time.

Future directions include an assessment of the trade-off between the cost of finding the undominated blocking pairs and that of treating larger neighborhoods. We also plan to apply a local search approach to other versions of the SMTI problem and to study other variant of our algorithm, for example including tabu search or other greedy heuristics.

References

1. Gusfield, D., Irving, R.W.: *The Stable Marriage Problem: Structure and Algorithms*. MIT Press, Boston (1989)
2. Gale, D., Shapley, L.S.: College admissions and the stability of marriage. *The American Mathematical Monthly* 69(1), 9–15 (1962)
3. Manlove, D., Irving, R.W., Iwama, K., Miyazaki, S., Morita, Y.: Hard variants of stable marriage. *Theor. Comput. Sci.* 276(1-2), 261–279 (2002)
4. Gent, I.P., Prosser, P.: An empirical study of the stable marriage problem with ties and incomplete lists. In: *ECAI*, pp. 141–145 (2002)
5. Gent, I.P., Prosser, P.: Sat encodings of the stable marriage problem with ties and incomplete lists. In: *SAT 2002*, pp. 133–140 (2002)
6. Yanagisawa: *Approximation algorithms for stable marriage problems*. PhD thesis, Kyoto University, Graduate School of Informatics (2007)
7. McDermid, E.: A $3/2$ -approximation algorithm for general stable marriage. In: Albers, S., et al. (eds.) *ICALP 2009, Part I*. LNCS, vol. 5555, pp. 689–700. Springer, Heidelberg (2009)
8. Iwama, K., Manlove, D., Miyazaki, S., Morita, Y.: Stable marriage with incomplete lists and ties. In: Wiedermann, J., Van Emde Boas, P., Nielsen, M. (eds.) *ICALP 1999*. LNCS, vol. 1644, pp. 443–452. Springer, Heidelberg (1999)
9. Holger, H., Hoos, E.T.: Local search methods. In: Rossi, F., Beek, P.V., Walsh, T. (eds.) *Handbook of Constraint Programming*. Elsevier, Amsterdam (2006)
10. Stützle, T.G.: *Local Search Algorithms for Combinatorial Problems - Analysis, Improvements, and New Applications*. PhD thesis, Am Fachbereich Informatik der Technischen Universität Darmstadt (1998)
11. Hoos, H.: On the run-time behaviour of stochastic local search algorithms for sat. In: *Proc. AAAI 1999*, pp. 661–666 (1999)
12. Roth, A.E., Vate, J.H.V.: Random paths to stability in two-sided matching. *Econometrica* 58(6), 1475–1480 (1990)

Layered Hypernetwork Models for Cross-Modal Associative Text and Image Keyword Generation in Multimodal Information Retrieval

Jung-Woo Ha, Byoung-Hee Kim, Bado Lee, and Byoung-Tak Zhang

Biointelligence Lab, School of Computer Science and Engineering,
Seoul National University,
599 Gwanak-ro, Gwank-gu, Seoul 151-744, Korea
{jwha, bhkim, bdlee, btzhang}@bi.snu.ac.kr

Abstract. Conventional methods for multimodal data retrieval use text-tag based or cross-modal approaches such as tag-image co-occurrence and canonical correlation analysis. Since there are differences of granularity in text and image features, however, approaches based on lower-order relationship between modalities may have limitations. Here, we propose a novel text and image keyword generation method by cross-modal associative learning and inference with multimodal queries. We use a modified hypernetwork model, i.e. layered hypernetworks (LHNs) which consists of the first (lower) layer and the second (upper) layer which has more than two modality-dependent hypernetworks and one modality-integrating hypernetwork, respectively. LHNs learn higher-order associative relationships between text and image modalities by training on an example set. After training, LHNs are used to extend multimodal queries by generating text and image keywords via cross-modal inference, i.e. text-to-image and image-to-text. The LHNs are evaluated on Korean magazine articles with images on women fashions and life-style. Experimental results show that the proposed method generates vision-language cross-modal keywords with high accuracy. The results also show that multimodal queries improve the accuracy of keyword generation compared with uni-modal ones.

Keywords: hypernetwork, layered hypernetwork, cross-modal generation, vision-language, text-to-image, image-to-text, multimodal information retrieval.

1 Introduction

Recently, cross-modal learning methods have been considered as a major approach for multimodal information retrieval such as video, image, and article retrieval as well as automatic tagging and annotation [1-3]. Because there are differences of granularity in text and image features, however, simple approaches based on text-image relations have the limitation to learn. As a model to learn higher-order cross-modal associations, we used hypernetwork models in the previous study [4]. A hypernetwork is a higher-order probabilistic graphical model which has properties including globality, compositionality, self-assembly, and recall-memory [5]. In the previous study, we

showed that images could be retrieved with multimodal queries by text-to-image inference with trained hypernetworks [4].

In this study, we propose a novel modified hypernetwork model, layered hypernetworks (LHNs), which conducts cross-modal associative learning and inference including image-to-text as well as text-to-image for multimodal information retrieval. An LHN is a hypernetwork model with a hierarchical structure of two layers of hypernetwork. While the first layer is composed of modality-dependent hypernetworks, only one hypernetwork exists in the second layer which represents relationships between the text modality and the image modality. The hierarchical structure makes LHNs analyzed with efficiency compared with conventional hypernetworks. Trained LHNs can generate both text and image keywords by cross-modal associative inference with multimodal queries. In addition, generated visual and textual keywords are used to retrieve articles by comparing them with text terms in document and visual words in images of articles. We use 983 Korean magazine articles with 8,763 images on women fashion and life-style as multimodal data. In this study, our contributions are summarized as follows.

1. We propose a novel modified hypernetwork named to layered hypernetwork for cross-modal associative learning and inference.
2. We propose a method to generate visual and textual keywords based on text-to-image and image-to-text cross-modal association.
3. We apply the proposed model to magazine article retrieval.

The rest of this paper is organized as follows. In Section 2, we summarize related works. Also, we explain layered hypernetworks for cross-modal association in Section 3 and propose a method for cross-modal keyword generation in Section 4. Section 5 presents the experimental results. Finally, we present concluding remarks in Section 6.

2 Related Works

As multi-media data increase explosively, multimedia data retrieval has been important problem in information retrieval such as video, image and articles. As an approach, cross-modal associative learning has been applied to multimodal data retrieval although cross-modal learning is from cognitive science and neuroscience [6]. Snoek *et al.* proposed concept-based video retrieval method [7] and Yan *et al.* studied a multimodal retrieval approach including text and image for broadcast new video [8]. D. Li *et al.* [9] suggested cross-modal association based factor analysis method as alternatives to Latent Semantic Indexing (LSI) and Canonical Correlation Analysis (CCA). Ferecatu *et al.* showed that the joint use of visual features and concept-based features with relevance feedback scheme improves the quality of the cross-modal image retrieval [10]. Goh *et al.* proposed an image retrieval method based on multimodal concept-dependent active learning [2]. Also, auto-annotation on unlabeled images and objects in images is carried out by using hierarchical latent Dirichlet allocation model [11]. In addition, human-computer interaction (HCI) is a research where cross-modal learning is considered as an essential element. In HCI, various modalities are studied including speeches and gestures. Quek *et al.* studied multimodal human

discourse in aspect of gesture and speech [12]. Christoudias *et al.* proposed co-training method of multimodal data to construct multimodal interface [13]. However, conventional studies on cross-modal learning are usually based on lower-order co-occurrence on modalities rather than higher-order relations. Therefore, we propose a cross-modal learning method based on higher-order inter-modal relationships in this paper.

3 Cross-Modal Associative Learning Models

3.1 Hypernetwork Model

A hypernetwork is a bio-inspired probabilistic graphical model based on hypergraph models. The properties of the hypernetwork model are summarized as three aspects: glocality, compositionality and self association based on randomness and recall [5].

1. Glocality: A hypernetwork consists of hyperedges with various orders. Lower-order hyperedges can represent general information and higher-order ones include more specific and local information.
2. Compositionality: A hypernetwork represents a huge structured combinatorial space. By learning based evolutionary strategy, a hypernetwork explores the combinatorial problem space.
3. Self association: The structure of hypernetworks is self-organized by evolutionary computation based on random selection. Self association makes the hypernetwork act like a recall memory.

Formally, a hypernetwork H is defined as $H = (V, E, W)$ where V , E , and W are a set of vertices, hyperedges, and weights. In hypernetworks, a vertex means a value of attributes and a hyperedge represents the combination of more than two vertices with its own weight. The number of vertices in a hyperedge is called cardinality or order of a hyperedge and k -hyperedge denotes a hyperedge with k vertices. When orders of all hyperedges are k , we call it k -hypernetwork. Therefore hypernetworks can represent higher-order relationships among large numbers of attributes.

Since a hypernetwork can be regarded as a probabilistic associative memory model to store segments of a given data set $D = \{\mathbf{x}^{(n)}\}_{n=1}^N$ i.e. $\mathbf{x} = \{x_1, x_2, \dots, x_m\}$, a learned hypernetwork can retrieve a data sample later. When $I(\mathbf{x}^{(n)}, E_i)$ denotes a function which yields the combination or concatenation of elements of E_i as (2), then, the energy of hypernetwork is defined as follows:

$$\mathcal{E}(\mathbf{x}^{(n)}; W) = - \sum_{i=1}^{|E|} w_i^{(k)} I(\mathbf{x}^{(n)}, E_i), \quad (1)$$

$$I(\mathbf{x}^{(n)}, E_i) = x_{i1}^{(n)} x_{i2}^{(n)} \dots x_{ik}^{(n)}, \quad (2)$$

where $w_i^{(k)}$ is a weight of i -th hyperedge E_i with k -order, $\mathbf{x}^{(n)}$ means the n -th stored pattern of data and E_i is $\{x_{i1}, x_{i2}, \dots, x_{ik}\}$. Then, the probability of the data generated by a hypernetwork $P(D|W)$ is given as a Gibbs distribution:

$$P(D|W) = \prod_{n=1}^N P(\mathbf{x}^{(n)} | W), \quad (3)$$

$$P(\mathbf{x}^{(n)} | W) = \frac{1}{Z(W)} \exp(-\mathcal{E}(\mathbf{x}^{(n)}; W)), \quad (4)$$

where $Z(W)$ is a partition function. In addition, the partition function $Z(W)$ is formulated as follow:

$$Z(W) = \sum_{\mathbf{x}^{(m)} \subset D} \exp \left\{ \sum_{i=1}^{|E|} w_i^{(k)} I(\mathbf{x}^{(m)}, E_i) \right\}. \quad (5)$$

That is, a hypernetwork is represented with a probability distribution of combination of variables with weights as parameters when we consider attributes in data as random variables. Considering that learning of hypernetworks is selecting hyperedges with high weight value, the learning can be considered as the process for maximizing log-likelihood. Learning from data is regarded as maximizing probability of weight parameter of a hypernetwork for given data. Given data, probability of a weight set of hyperedges $P(W|D)$ is defined as follows:

$$P(W|D) = \frac{P(D|W)P(W)}{P(D)}. \quad (6)$$

According to (4) and (6), then, likelihood is defined as

$$\prod_{n=1}^N P(\mathbf{x}^{(n)} | W)P(W) = \left(\frac{P(W)}{Z(W)} \right)^N \exp \left\{ -\sum_{n=1}^N \mathcal{E}(\mathbf{x}^{(n)} | W) \right\}. \quad (7)$$

Ignoring $P(W)$, maximizing the argument of exponential function is obtaining maximum likelihood. Using log function,

$$\arg \max_W \left[\log \left\{ \prod_{n=1}^N P(\mathbf{x}^{(n)} | W) \right\} \right] = \arg \max_W \left\{ \sum_{n=1}^N \sum_{i=1}^{|E|} w_i^{(k)} I(\mathbf{x}^{(n)}, E_i) - N \log Z(W) \right\}. \quad (8)$$

More explanations on the derivative of the log-likelihood are showed in [5]. Therefore, log-likelihood of hypernetwork can be maximized by decreasing the difference of hyperedges from a given data set.

3.2 Layered Hypernetworks

An LHN is a hypernetwork with hierarchical structures and the model consists of two layers. The first layer is a modality layer and the second one is an integrating layer. When data consisting of more than one modality are given, the attributes of given data are partitioned based on modalities. Hypernetworks in the first layer are built by sampling from attributes of each modality and the number of hypernetwork in the first layer is equal to the number of modalities. Dissimilar to the first layer, only one hypernetwork exists in the second layer. The second layer hypernetwork is built by combining hyperedges randomly selected from modality-dependent hypernetworks in

the first layer. Therefore the hypernetwork in the second layer represents the relationship between several modalities. Same as conventional hypernetworks, formally, the second-layer hypernetwork is defined with the energy function when a weight vector is given as a parameter. When given a data set D consisting of two modalities, $D = \{\mathbf{x}^{(n)}\}_{n=1}^N = \{(\mathbf{m}^1, \mathbf{m}^2)^{(n)}\}_{n=1}^N$, the energy of the second-layer hypernetwork $\mathcal{E}(\mathbf{x}^{(n)}; W)$ generated from k -hypernetworks in the first-layer is defined as follows:

$$\mathcal{E}(\mathbf{x}^{(n)}; W) = \mathcal{E}\{(\mathbf{m}^1, \mathbf{m}^2)^{(n)}; W\} = -\sum_{i=1}^{|E|} w_i^{(k)} I\{(\mathbf{m}^1, \mathbf{m}^2)^{(n)}, E_i\}, \quad (9)$$

where \mathbf{m}^1 and \mathbf{m}^2 are vectors of each modality variable which constitute the n -th data sample $\mathbf{x}^{(n)}$. Same as (4), then, the probability of generating n -th data with two modalities, $P(\mathbf{x}^{(n)}|W)$ is defined as follows:

$$P(\mathbf{x}^{(n)}|W) = \frac{1}{Z(W)} \exp\left[-\mathcal{E}\{(\mathbf{m}^1, \mathbf{m}^2)^{(n)}; W\}\right]. \quad (10)$$

Assuming that \mathbf{m}^1 , \mathbf{m}^2 are text and image modality respectively, similar as conventional hypernetwork, the probability of data generated by layered hypernetworks, $P(D|W)$ is defined as follows:

$$\begin{aligned} P(D|W) &= P(T, I|W) = P(T|I, W)P(I|W) \\ &= P(I|T, W)P(T|W). \end{aligned} \quad (11)$$

Formula (11) means that cross-modal inferences between text and image are carried out by learning parameters of hypernetworks. Figure 1 shows the architecture of LHNs.

3.3 Cross-Modal Associative Learning of Layered Hypernetworks

3.3.1 Learning of the First-Layer Hypernetworks

Learning of the first-layer hypernetworks is similar to the learning of conventional hypernetworks [4-5] except building a hypernetwork per one modality. At first, multimodal data are separated by modalities. In this study, an article data with unique id are divided into vectors of TF-IDF values from documents and vectors of histogram value from included images. The unique id is used to combine hyperedges of each modality in learning of the second-layer hypernetwork. Building a hypernetwork is carried out by generating hyperedges from each modality and hyperedges are generated by selecting and combining the attributes with non-negative values with randomness for each modality. The reason to select the attributes with non-negative values is that hyperedges where values of all vertices are zero may be generated with high probability because most attributes have zero value due to sparsity of data. As explained in Section 3, learning of hypernetwork is sampling hyperedges which are less different from data set. Details of building and learning a hypernetwork are explained in [5]. As learning continues, the structure of a hypernetwork fits the distribution of given data more. The constitution of hyperedges, the structure of a hypernetwork, is

determined by their weights which reveal the fitness with training data set. In this study, we define the weight of a hyperedge, w , as follows:

$$w = \frac{C}{\# \text{ of matched training samples} + k}, \tag{12}$$

where k denotes order of a hyperedge and C is an arbitrary constant. According to (12), hyperedges with unique information get higher weights by definition. Also, hyperedges with low weight values are eliminated and the erased amounts of hyperedges are regenerated from training set.

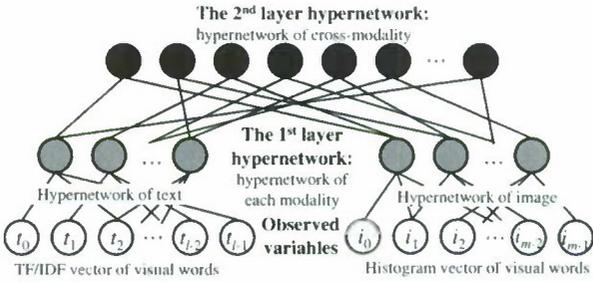


Fig. 1. Architecture of layered hypernetwork models

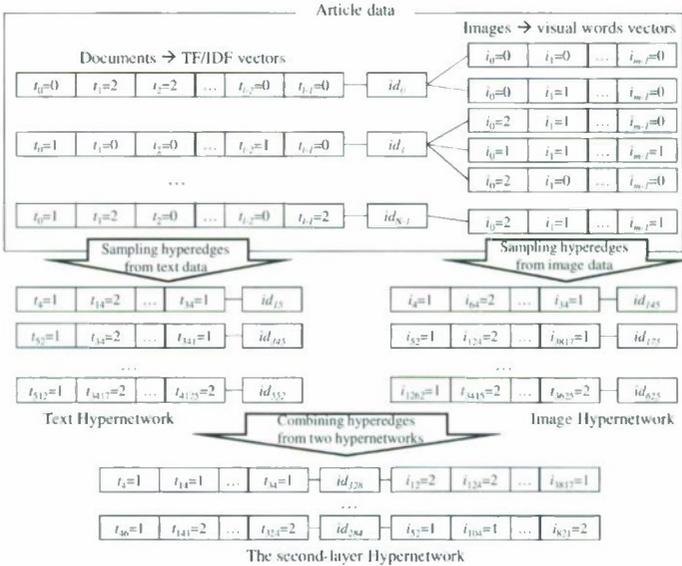


Fig. 2. The process of making and learning a layered hypernetwork

3.3.2 Learning of the Second-Layer Hypernetwork

Learning of the second-layer hypernetwork is to generate hyperedges which represent high-order relationships between modalities from the first-layer hypernetworks. Hyperedges of the second-layer hypernetwork are generated by combining hyperedges of hypernetworks in the first-layer. In combining, hyperedges from different modalities with the same id are merged into a new hyperedge. The weight of the generated hyperedge is obtained by comparing with training set same and hyperedges with low weights are also eliminated from the hypernetwork same as learning in the first layer. Then, the generated hypernetwork is evaluated with training data set. Figure 2 shows the process of making and learning a layered hypernetwork. In addition, algorithm of building and learning the second-layer hypernetwork is presented in detail in Figure 3. In our method, learning process finishes after fixed number of epochs.

```

HT: hypernetwork from text data, HI: hypernetwork from image data,
HL: layered hypernetwork R: replacing rate of hyperedges with low weights
CR: combining rate of hyperedges of HT with a hyperedge of HI
HT ← makeHypernetwork(T); HI ← makeHypernetwork(I)
For i ← 1 until end condition
  HT ← learningHypernetwork(T); HI ← learningHypernetwork(I);
  HT ← removeLowedges(R); HI ← removeLowedges(R); HL ← {};
  For j ← 1 to |HT|
    ET ← the j-th hyperedge of HT
    For k ← 1 to CR
      EI ← a randomly selected hyperedge with same id to ET from HI;
      EL ← ET ∪ EI; HL ← HL ∪ EL
    End For
  End For
  HL ← removeLowedges(R); HL ← learningHypernetwork(T, I);
  evaluate(HL, I, T)
  HT = Resampling(T, R); HI = Resampling(I, R)
End For

```

Fig. 3. Algorithm of building and learning a layered hypernetwork. Details of functions for learning are explained in our previous studies [4-5].

4 Cross-Modal Inference for Image and Text Keyword Generation

Trained LHNs can generate both text terms and visual words with given multimodal queries by cross-modal associative inference. Cross-modal associative generation is divided into two types such as text-to-image to generate a set of visual words for given text terms and image-to-text generation to reconstruct a set of text terms with visual words. In image-to-text, the generated set of text terms is composed of text terms in hyperedges of the second-layer hypernetwork whose vertices include at least one visual word in the given set of visual words. To select text terms, we define a

score based on co-occurrence of text terms and visual words. When a visual word set Q , the score $s_{Idx(i),E_n}$ of the i -th text term in the n -th hyperedge E_n of the second-layer hypernetwork is defined as follow:

$$s_{Idx(i),E_n} \begin{cases} \frac{x_{Idx(i)}^2 \times w_n}{|Q - E_n| \times C + 1} & (Q \cap E_n \neq \emptyset) \\ 0 & (Q \cap E_n = \emptyset) \end{cases}, \quad (13)$$

where $x_{Idx(i)}$ is the value of text term attribute whose index is $Idx(i)$, $Idx(i)$ denotes the index in the vector representation of the i -th text term of a hyperedge E_n , w_n means weight of E_n , $|Q - E_n|$ is the size of the relative complement, and C is a arbitrary constant for penalty. Therefore, $s_{Idx(i)}$ is obtained by summing for all hyperedges as follow:

$$s_{Idx(i)} = \sum_{n=1}^{|E|} s_{Idx(i),E_n}, \quad (14)$$

where $|E|$ denotes the number of hyperedges in the second-layer hypernetwork. According to (13), as a hyperedge includes more visual words in given visual word set, the score of text terms in the hyperedge gets larger. Then, text terms with higher score are included candidates for generated text keywords.

Same as image-to-text, a set of visual words are generated with trained layered hypernetwork and given text terms.

5 Experimental Results

5.1 Data and Experimental Setups

We use 983 articles with 8,673 images from three Korean magazines on female fashion name to 'luxury', 'beauty life' and 'haute' respectively as training data from a company named to ddh co. As preprocessing for modeling, documents in articles are converted to vectors of TF-IDF values of 5,000 text terms which are selected by

Table 1. The parameters used for the experiment

Parameters	Value
Order (text, image)	(20, 20)
Replacing rate	0.1
Sampling rate (text, image)	(20, 10)
Combining rate	10
Num. of iteration	5

Combining rate means the combining number of hyperedges of one modality hypernetwork for a hyperedge of the other modality hypernetwork in learning of the second layer. Sampling rate denotes the size of sampled hyperedges from a training data sample. Replacing rate is eliminated ratio of hyperedges with low weight in one iteration.

occurrence frequency in documents after stemming. Also, an image is represented with a vector of histograms of 4,022 visual words extracted by SURF [14]. Then, values of each modality are converted to three-level values from 0 to 2 since hypernetwork models can deal with discretized data. Data are divided into a training set with 884 documents and 7,555 images and a test set consisting of 99 documents and 845 images for article retrieval. Table 1 shows the parameter setting to train layered hypernetworks.

5.2 Experimental Results

We evaluate the similarity of cross-modal associative generation by comparing generated text terms and visual words with text and image keywords in the given query. To evaluate the similarity, we define two measures in this paper. The first measure is ratio of correctness (RC). Referring a set whose elements are text terms and visual words which constitute a document and an image in an article to an original set, we generate text terms or visual words as same amount as the size of the original set. Then we compare a generated textual or visual set with the original set when partial text terms and visual words are given. RC is defined as follow:

$$RC = \frac{\# \text{ of generated keywords same to keywords in an original set}}{\# \text{ of generated text (image) keywords}}. \quad (15)$$

According to (15), RC can have a value from 0 to 1. The second measure is context score (CS) which are based on pair-wise co-occurrence of all text terms and visual words with non-negative value in documents and images of article data. To obtain CS, we define a measure of pair-wise co-occurrence for the i -th and j -th keyword as follow:

$$m_{ij} \begin{cases} \sum_{n=1}^N \frac{x_i^{(n)} \times x_j^{(n)}}{\{(x_i^{(n)})^2 - (x_j^{(n)})^2\}^2 + 1} & (i \neq j) \\ C & (i = j) \end{cases}, \quad (16)$$

where x_i and x_j is the value whose indices are i and j in the n -th data sample $\mathbf{x}^{(n)}$, N is the size of data set, and C is a arbitrary constant. Then, CS is defined as follow:

$$CS = \frac{1}{|G|} \sum_{i,j} m_{ij}, \quad (17)$$

where $|G|$ is the size of set of generated text terms or visual words. The different point of CS from RC is that CS reflects the contexts of relationships between generated keywords. Although RCs of two generated sets are same, CSs may be different each other dependent on the co-occurrence frequency of wrongly generated keywords. Figure 4 and 5 are the result of text-to-image generating visual words and image-to-text generating text terms for all training data when a few text terms and visual words are given as a query. Figure 4 shows average RC and CS of generated text terms by image-to-text generation for 889 documents. Cross-modal queries can improve more 40% point of accuracy of the generation of text terms related to given queries

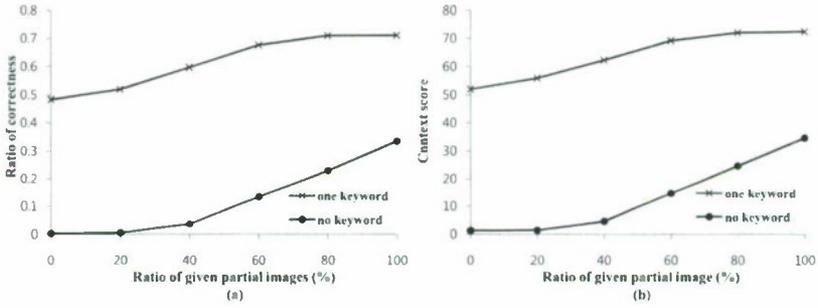


Fig. 4. Average RC (a) and CS (b) of generated visual words by image-to-text generation

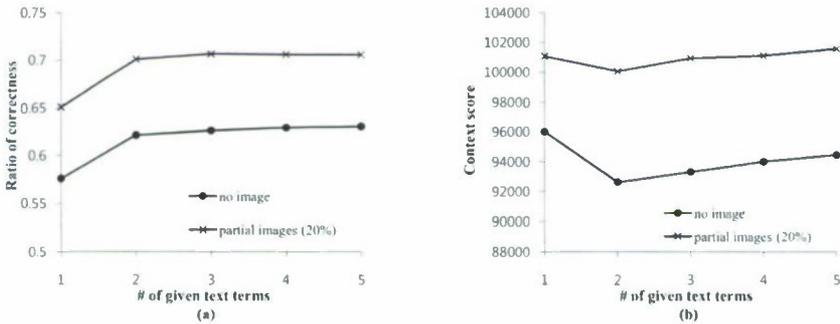


Fig. 5. Average RC (a) and CS (b) of generated keywords by text-to-image generation. Scale of context score of text to image generation is much larger than one of text-to-image generation since the size of image data is approximately ten times and non-zero variables in histogram vector of images are much more than in TF-IDF vector of documents.

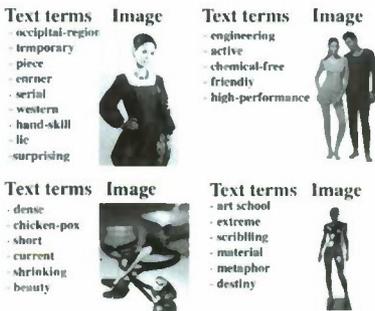


Fig. 6. Articles whose text terms are generated perfectly with given one text term and 20% of visual words in the article

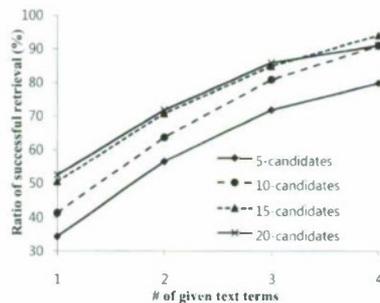


Fig. 7. Ratio of successful retrievals for test data set as the number of given text terms increases

compared with text query only. According to Figure 4, when the same amount of text terms is given, the similarity score of generated text terms get higher as information of given image increase. Also, without any text keyword query, text terms in the original set can generated with partial images only. Figure 5 presents average RC and CS of generated visual words by text-to-image generation for 884 images among training images. Same as Figure 4, multimodal information increases two scores compared with image input only. Dissimilar to image-to-text generation, RCs are saturated when more than two text terms are given. In addition, CSs show different patterns from image-to-text generations. It is the reason that an article consists of one document and several images so that image information is more important than text information. Figure 6 shows four pairs of the set of text terms and an image of articles whose RCs are 1 when one text terms and 20% of visual words in the article are given as a query. We can generate text terms and retrieve the article with small part of information by cross-modal associative generation. Figure 7 presents the ratio of successful article retrieval when partial text terms of a data are given for test data set using trained layered hyper-network. In this study, article retrieval is considered to be successful when candidates include the test article whose text terms and visual words are given as a query. According to Figure 7, with both more than two text terms and half of image, the article which a user wants can be included over 90% when the size of candidates is 20.

6 Concluding Remarks

In this paper, we propose LHNs for cross-modal associative learning and a method to generate visual and textual keywords based on text-to-image and image-to-text cross-modal inference with LHNs for given multi-modal queries. Experimental results show that it is possible to generate keywords based on cross-modal association of inter-modalities. Also, multimodal queries improve the similarity of generated keywords compared with uni-modal ones. In addition, we show that proposed model and method can be applied to an articles retrieval system. As future works, we will apply the cross-modal associative keyword generation method to various problems such as auto-annotation for unlabeled images as well as multimodal information retrieval.

Acknowledgements

This work was supported in part by IT R&D Program of MKE/KEIT (KI002138, MARS), in part by NRF Grant of MEST (314-2008-1-D00377, Xtran), the National Research Foundation of Korea(NRF) grant funded by the Korea government(MEST) (No. 2010-0017734), and in part by the BK21-IT program funded by Korean Government (MEST).

References

- [1] Datta, R., Joshi, D., Li, J., Wang, J.Z.: Image retrieval: Ideas, influences, and trends of the new age. *ACM Computing Surveys (CSUR)*, Article 5, 40(2) (2008)
- [2] Goh, K.-S., Chang, E.Y., Lai, W.-C.: Multimodal concept-dependent active learning for image retrieval. In: *Proc. of the 12th Annual ACM International Conference on Multimedia (MM 2004)*, pp. 564–571 (2004)

- [3] Simon, I., Snavely, N., Seitz, S.M.: Scene Summarization for Online Image Collections. In: Proc. of 11th IEEE International Conference on Computer Vision, ICCV 2007 (2007)
- [4] Ha, J.-W., Kim, B.-H., Kim, H.-W., Yoon, W.C., Eom, J.-H., Zhang, B.-T.: Text-to-image cross-modal retrieval of magazine articles based on higher-order pattern recall by hypernetworks. In: Proc. of the 10th International Symposium on Advanced Intelligent Systems (ISIS 2009), pp. 274–277 (2009)
- [5] Zhang, B.-T.: Hypernetworks: A molecular evolutionary architecture for cognitive learning and memory. *IEEE Computational Intelligence Magazine* 3(3), 49–63 (2008)
- [6] Fuster, J.M., Bodner, M., Kroger, J.K.: Cross-modal and cross-temporal association in neurons of frontal cortex. *Nature* 405, 347–351 (2000)
- [7] Snock, C.G.M., Worring, M.: Concept-based video retrieval. *Foundations and Trends in Information Retrieval* 2(4), 215–322 (2009)
- [8] Yan, R., Hauptmann, A.G.: A review of text and image retrieval approaches for broadcast news video. *Information Retrieval* 10(4-5), 445–484 (2007)
- [9] Li, D., Dimitrova, N., Li, M., Sethi, K.: Multimedia content processing through cross-modal association. In: Proc. of the 11th Annual ACM International Conference on Multimedia (MM 2003), pp. 604–611 (2003)
- [10] Frecatu, M., Boujemaa, N., Crucianu, M.: Semantic interactive image retrieval combining visual and conceptual content description. *Multimedia Systems* 13, 309–322 (2008)
- [11] Yakhnenko, O., Honavar, V.: Annotating images and image objects using a hierarchical dirichlet process model. In: Proc. of the 9th International Workshop on Multimedia Data Mining in ACM SIGKDD 2009, pp. 1–7 (2009)
- [12] Quek, F., McNeil, D., Bryll, R., Duncan, S., Ma, X.-F., Kirbas, C., McCullough, K.E., Ansari, R.: Multimodal human discourse: gesture and speech. *ACM Trans. on Computer-Human Interaction* 9(3), 171–193 (2002)
- [13] Christoudias, C.M., Saenko, K., Morency, L.-P., Darrell, T.: Co-Adaptation of audio-visual speech and gesture classifiers. In: Proc. of the 8th International Conference on Multimodal Interfaces, pp. 84–91 (2006)
- [14] Bay, H., Tuytelaars, T., Gool, T.V.: Surf: Speed up robust features. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) *ECCV 2006*. LNCS, vol. 3951, pp. 404–417. Springer, Heidelberg (2006)

Visual Query Expansion via Incremental Hypernetwork Models of Image and Text

Min-Oh Heo, Myunggu Kang, and Byoung-Tak Zhang

Biointelligence Lab, School of Computer Science and Engineering,
Seoul National University,
599 Gwanak-ro, Gwank-gu, Seoul 151-744, Korea
{moheo, mgkang, btzhang}@bi.snu.ac.kr

Abstract. Humans can associate vision and language modalities and thus generate mental imagery, i.e. visual images, from linguistic input in an environment of unlimited inflowing information. Inspired by human memory, we separate a text-to-image retrieval task into two steps: 1) text-to-image conversion (generating visual queries for the 2 step) and 2) image-to-image retrieval task. This separation is advantageous for inner representation visualization, learning incremental dataset, using the results of content-based image retrieval. Here, we propose a visual query expansion method that simulates the capability of human associative memory. We use a hypernetwork model (HN) that combines visual words and linguistic words. HNs learn the higher-order cross-modal associative relationships incrementally on a set of image-text pairs in sequence. An incremental HN generates images by assembling visual words based on linguistic cues. And we retrieve similar images with the generated visual query. The method is evaluated on 26 video clips of ‘Thomas and Friends’. Experiments show the performance of successive image retrieval rate up to 98.1% with a single text cue. It shows the additional potential to generate the visual query with several text cues simultaneously.

Keywords: hypernetwork, incremental data, visual query expansion, vision-language, text-to-image, multimodal information processing.

1 Introduction

Conventional text-to-image retrieval methods for image-text corpus have used the annotated tags on images that are used for searching for the target [1]. Recently, multi-modal data such as video, sound, images as well as web-pages including images are increasing explosively. Consequently, the underlying data distribution may change over time [3]. So, we need incremental models to learn the data of multi-modality.

Humans can associate vision and language modalities and thus generate mental imagery, i.e. visual images, from linguistic input in the environment of unlimited inflowing information. Considering human capability of multimodal memory [2,5,16], we separate a text-to-image retrieval task into two steps. In the first step, text-to-image conversion is used to generate the visual concept from the related

images associated with text cues. And the second step is to search for similar images with the expanded visual query from the first step. This approach gives some advantages. First, we can visualize the inner representation of the form of visual images. Secondly, we can deal with incremental data by updating visual queries incrementally in the first step. Thirdly, we can bring the result from content-based image retrieval (CBIR) for the second step. In addition, after generating visual queries with enough large data, we expect the visual queries to be the universal visual concepts when retrieving from all image databases.

Here, we propose a novel visual query expansion method that simulates the capability of human associative memory. Hypernetwork models (HN) have cognitive properties of continuity, glocality, and compositionality [5]. And HNs learn higher-order cross-modal association to solve the difference of granularity in image and text features. HNs can be appended and updated partially by adding new hyperedges from new observations as incremental learning. Especially, we built a visual word dictionary keeping the regional information from an image beforehand. This enables us to visualize the visual query and avoid the limitation of computational complexity for the image representation. As Fig. 1 shows, 1007 image-text pairs were captured from 26 video clips of Thomas and Friends. And we simply used the sum of absolute difference in RGB scale between images as the second step.

This paper is organized as follows. Section 2 summarizes related works. Then hypernetworks will be introduced briefly in Section 3 and a proposed method is explained in Section 4. Section 5 shows the experimental results. Finally, Section 6 concludes this paper with concluding remarks.

2 Related Work

Crossmodal data retrieval has been focused on the information retrieval field, as a result of readily available multimedia data. Approaches using multimodal data have been introduced using tagging based methods such as automatic tagging and annotation and statistical dependency based methods such as co-occurrence and canonical correlation analysis (CCA) [1-2]. And approaches using image annotation

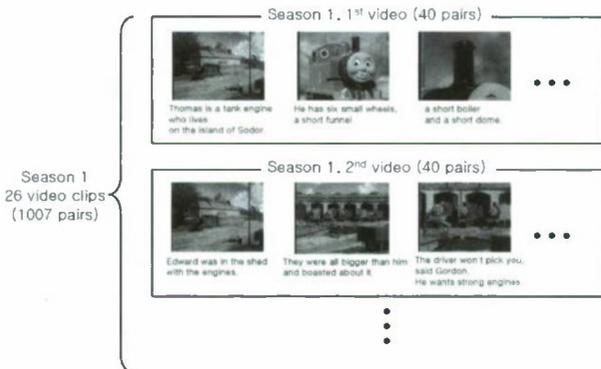


Fig. 1. The training dataset used in this paper. The pairs from one clip are one unit of instances for sequential presentation on incremental learning.

information were studied. Jeon *et al.* proposed a cross-media relevance model (CMRM) [6] using annotated images and grouping small blobs of images manually. And Pan *et al.* studied graph-based methods for the correlated nodes discovery across other modalities [7]. And cross-modal association learning has been applied to video data. Yan *et al.* studied a text-image multimodal retrieval task on data of a broadcast new video [9] and Snoek *et al.* suggested a concept-based video retrieval method [8]. Additionally, D. Li *et al.* proposed a factor analysis method based on cross-modal association [10].

For the visual query expansion, it is mainly used to improve the performance of the retrieval task. Chum *et al.* introduced query expansion using images by analogy for the text retrieval. They used images as added queries giving spatial constraints and improved the retrieval performance for false negatives [12]. Joly *et al.* applied this concept to logo retrieval in large image collection [13] and Jiang *et al.* did this to bag-of-visual-words [14]. As visual representational aspects, a visual mental imagery is used as inner representation of cognitive processes of humans [16], AIs [17] and even robots [18].

In [4], Ha *et al.* studied the image-text cross-modal retrieval task with multimodal queries based on pixels of the gray scale on the fixed dataset. On the contrary, we deal with the relevant image retrieval task based on incremental HNs with color image patches on the increasing dataset.

3 Multi-modal Hypernetwork Models

3.1 Hypernetwork Models

A hypernetwork (HN) is a hypergraph which is represented with vertices and weighted hyperedges. Hypergraphs refer to generalized simple graphs by allowing for edges of higher cardinality. The edges in a hypergraph are called hyperedges. Fig. 2 shows an example of HN. In formal definition, a HN is defined as $H = (V, E, W)$ where V, E and W are a set of vertices, hyperedges, and weights respectively. And the elements of W correspond to the elements of E . A HN is formulated on the basis of probabilistic theory. Given a data set $D = \{\mathbf{x}^{(n)}\}_{n=1}^N$ of N samples, the HN can be

$$P(D|W) = \prod_{n=1}^N P(\mathbf{x}^{(n)}|W) \quad (1)$$

$$P(\mathbf{x}^{(n)}|W) = \frac{1}{Z(W)} \exp(-\mathcal{E}(\mathbf{x}^{(n)}; W)) \quad (2)$$

where $Z(W)$ denotes the partition function as the normalization term and $\mathbf{x}^{(n)}$ means the n -th instance of data. And \mathcal{E} is the energy function of HN and the partition function are defined as

$$\mathcal{E}(\mathbf{x}^{(n)}; W) = -\sum_{m=1}^{|E|} w_m \delta(\mathbf{x}^{(n)}, E_m) \quad (3)$$

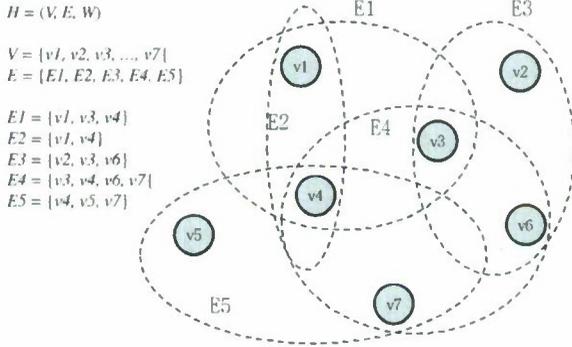


Fig. 2. An example of a hypernetwork. Hypernetwork H is composed of vertices set V , hyperedge set E and the corresponding weight W .

$$Z(W) = \sum_{n=1}^{|D|} \exp \left(- \sum_{m=1}^{|E|} -w_m \delta(\mathbf{x}^{(n)}, E_m) \right) \quad (4)$$

where $w_i^{(k)}$ is a positive real-valued weight of i -th hyperedge E_i and $\delta(\mathbf{x}^{(n)}, E_i)$ denotes the identity function depending on input parameter elements of $\mathbf{x}^{(n)}$ and hyperedge E_i .

Taking the derivative of log-likelihood function of (2), we can derive the following

$$\ln P(D|W) = \ln \prod_{n=1}^N P(\mathbf{x}^{(n)} | W) \quad (5)$$

$$\begin{aligned} \nabla_w \ln \prod_{n=1}^N P(\mathbf{x}^{(n)} | W) &= \nabla_w \left\{ \ln \prod_{n=1}^N \frac{1}{Z(W)} \exp(-\mathcal{E}(\mathbf{x}^{(n)}; W)) \right\} \\ &= N \left\{ \left\langle \delta(\mathbf{x}^{(n)}, E_m) \right\rangle_{Data} - \left\langle \delta(\mathbf{x}^{(n)}, E_m) \right\rangle_{P(\mathbf{x}|W)} \right\} \end{aligned} \quad (6)$$

And minimizing the difference between two average frequencies is equivalent to maximizing the likelihood by making (6) be equal to zero [5].

Then, the term

$$\sum_{n=1}^N \sum_{m=1}^{|E|} \delta(\mathbf{x}^{(n)}, E_m) = N \left\langle \delta(\mathbf{x}^{(n)}, E_m) \right\rangle_{Data} \quad (7)$$

can also be derived and it means that the total number of matching hyperedges with the given data set D follows the average frequencies of the hyperedges in the data set.

3.2 Cross-Modal Associative Learning on Incremental Hypernetwork Models

To learn cross-modal associative information, we create cross-modal hyperedges composed exclusively of the textual part and visual part, which are sampled from text and image respectively, as shown in Fig. 3. Formally, given an instance $\mathbf{x} = \{x_I, x_T\}$, x_I is the feature set for image representation and x_T is that for text representation:

$$X_I = \{x_1^i, x_2^i, x_3^i, \dots, x_P^i\} \tag{8}$$

$$X_T = \{x_1^j, x_2^j, x_3^j, \dots, x_Q^j\} \tag{9}$$

where P and Q are the number of features for images and text respectively, which means the size of visual word dictionary and linguistic word dictionary. x_k^i and x_j^j are features denoting the k -th element of the visual word dictionary and the j -th one of the text word dictionary respectively. Then the joint distribution given arbitrary weights from (1) can be converted using the composition of hyperedges, and written into the formulation taken from (7) by changing the weight reflecting the number of matched instances among the size N of dataset.

$$P(D|W) = P(D_I, D_T | W) \propto \sum_{n=1}^N \sum_{m=1}^{|E|} \delta(\mathbf{x}^{(n)}, E_m) = \sum_{m=1}^{|E|} w_m \delta(\mathbf{x}, E_m) \tag{10}$$

where D_I is the dataset of image features and D_T is the one of text features. Then, the distribution is represented by weighted nonzero basis functions having a zero-one binary value. However, all of the possible hyperedges from order 1 to the order of the number of total features is almost impossible by virtue of combinatorial explosion which dictates that the number of cases will massively increase. So, we should approximate this with the relatively small number of hyperedges by using random sampling strategy. We can approximate the joint distribution using M hyperedges like this formula,

$$P(D_I, D_T | W) \propto \sum_{m=1}^{|E|} w_m \delta(\mathbf{x}, E_m) \approx \sum_{m=1}^M w_m \delta(\mathbf{x}, E_m) \tag{11}$$

if M is large enough to express the distribution, the error between the estimation result and the distribution will be decreased. By this fact, we can estimate the distribution roughly by simply using a reasonably small number of hyperedges.

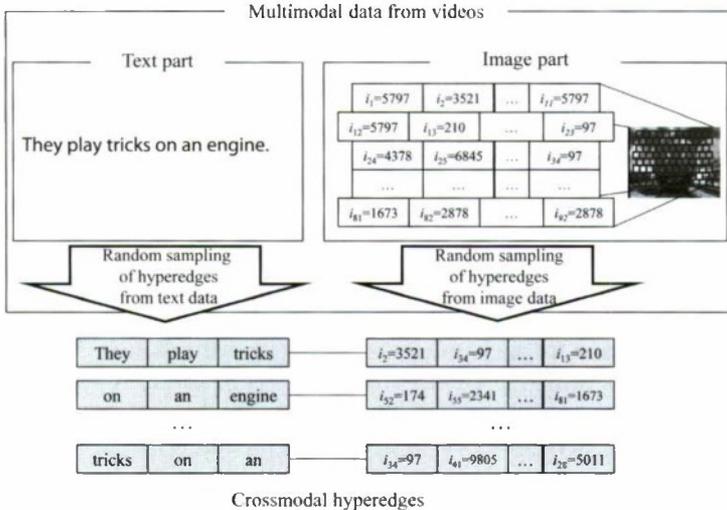


Fig. 3. An example of cross-modal hyperedges using the visual word dictionary. For the experiments, tri-gram is used for the sampling from text part and image patches random sampled among 92 regions on the grid for image part.

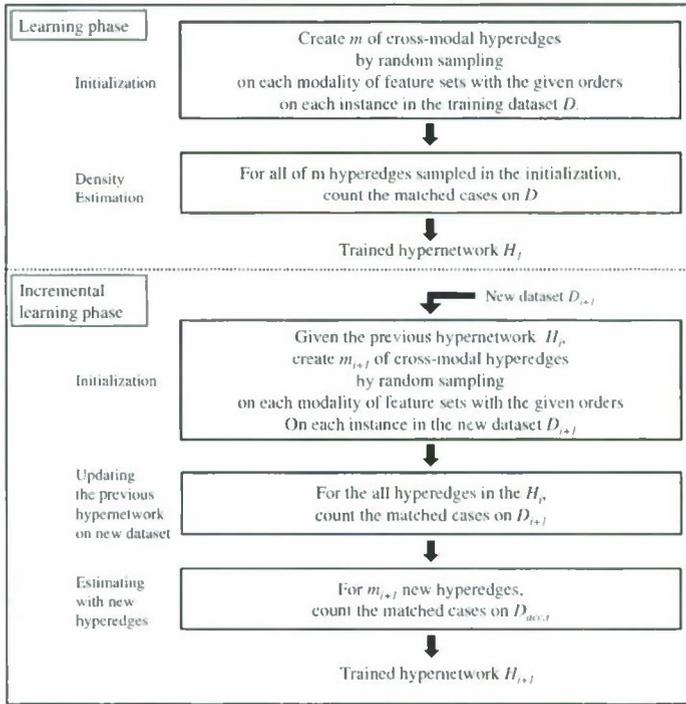


Fig. 4. The flow chart for cross-modal associative learning. The top shows the case for the fixed dataset and the bottom shows that for the incremental dataset.

For incremental HN learning, we can easily apply the same strategy with a small adjustment. Formally, we define the preliminary dataset as D_0 and the n -th new dataset as D_{n+1} . Then, the n -th accumulated training set $D^{(n+1)}$ can be written as follows:

$$D^{(n+1)} = D^{(n)} \cup D_{n+1} \tag{12}$$

Whenever there is an inflowing new dataset, adding new hyperedges from it by random sampling strategy can maintain the small error between the estimation and the distribution while keeping the condition that the number of hyperedges is enough to follow. The process is summarized in Fig. 4.

4 A Visual Query Expansion Method

4.1 Building a Visual Word Dictionary for Image Patches

Visual query expansion needs image processing for using visual features. Avoiding the vast computational complexity on the image representation, we built a visual word dictionary including 10,000 visual words beforehand. This process is illustrated in Fig. 5. As image preprocessing, each image is firstly segmented into 15×15 square image patches on a regular grid shown in the second image in Fig. 3. Following the

work of Feng *et al.* [15], using the rectangular regions could provide performance gains compared with using regions by automatic image segmentation methods. We were also able to avoid the problems associated with the computational cost. Secondly, we assigned all of the segmented patches into k groups by k -means clustering in the RGB color space using Koen’s image processing package [11]. As a result, we made 10,000 visual words by choosing the closest visual word from the centroid of each cluster. This set of image patches worked as visual words in this paper.

4.2 Visual Query Expansion by Combining Image Patches

Expanded Visual query can be created by the following process. When given the linguistic cue which works as the condition on the (10), we can make inference with the trained HN by the following formula

$$P(D_i | D_{T_q}, W) = \frac{P(D_i, D_{T_q} | W)}{P(D_{T_q} | W)} \propto \sum_{m \in E_{T_q}} w_m \delta(\mathbf{x}, E_m) \tag{13}$$

where the set E_{T_q} of cross-modal hyperedges including the text T_q . Then, we choose the index x_p^i of visual word that makes conditional likelihood be the maximum at the j -th region on the grid as follows:

$$I_j^{vq} = \arg \max_p P(I_j = f(x_p^i) | D_{T_q}, W) = \arg \max_p \sum_{m \in E_{T_q}} w_m \delta(\mathbf{x}, E_m) \tag{14}$$

where f is the mapping function to the visual word. And combining them generates visual query. This process can be achieved on HNs by choosing the visual word that maximum weight of hyperedges which are relevant to the text T_q as in the following summarized procedure.

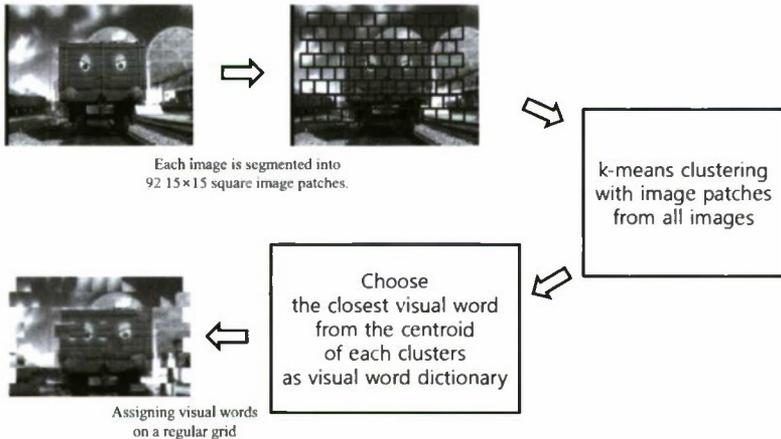


Fig. 5. The process to build a visual word dictionary and to convert original images into ones to be trained. All of the image patches segmented are grouped into 10,000 clusters and converted by the closest visual word from original image patches.

1. Summing up the weights of hyperedges having the text T_q .
2. Choose the index of visual word that make conditional likelihood be maximum at the j -th region on the grid.
3. Combining the image patches with the corresponding index at the j -th region.

5 Experimental Results

5.1 Data and Experimental Setups

As mentioned briefly in Section 1 and Fig. 1, we captured 1007 image-text pairs from 26 video clips of Thomas and Friends season 1. We used a capture tool to collect image-text pairs automatically whenever a subtitle appeared. Table 1 shows the distribution across 26 video clips. And the experimental setting is shown in Table 2.

5.2 Experimental Results

During the incremental learning, HNs were trained in sequence and retrieved top-N closest images using the sum of absolute difference in RGB scale between the generated visual query and original images to perform an image-to-image retrieval task. Fig. 6 and Fig. 7 show the results of image order 5 and order 35 each when the cue ‘engine’ is given. Then, shown in order, are the generated visual query, the closest top-5 images near the visual query in that dataset D_n and all of the original images associated with the cue in D_n . The associated original images are the same, but the generated visual queries are rather different, which cause the top-5 retrieved images to also be different. They include some original images (10/23 cases of nonzero original images, 10/45 in total). The visual queries generated from 5-order HNs are more flexible to incoming new instances than those by 35-order HNs. (25 consecutive difference $(\|V_i - I_{i,1}\|)$ per pixel: $\sigma_5 = 18.7 < \sigma_{35} = 26.6$, $m_5 = 17.1 \approx m_{35} = 18.7$).

In more than 2 words cases, the visual query can be generated. Fig. 8 shows a comparison between the case given text cues ‘noise’ and ‘once’ simultaneously and each. Though they are generated from the same HN model, at each case, they reflect the original images well, and the 2 words case do also. Even though there is no instance

Table 1. The frequencies of instances in the incremental dataset

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	Total
40	40	35	39	45	36	35	38	44	46	41	38	42	40	36	38	38	37	35	38	41	39	36	36	35	39	1007

Table 2. The information and parameter set for the experiments

Information	Values	Parameters	Values
Total data	1007 in 26 sets	Text order	3 (tri-gram)
Total text words	1256	Image order	5, 35
Number of regions on 1 image	92	Sampling rate	10
Number of visual words	10,000	Image patch size	15 × 15

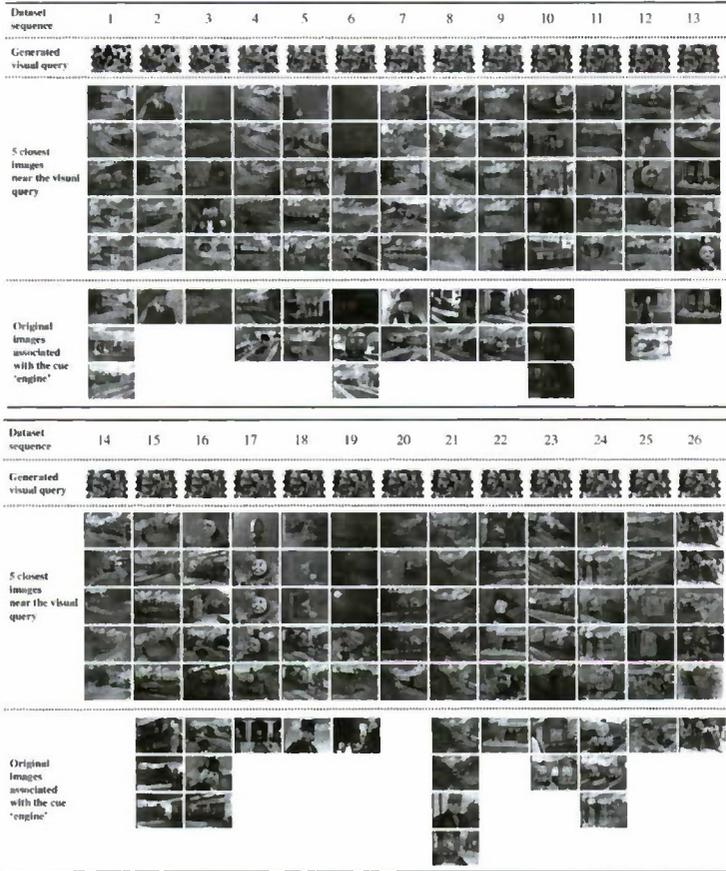


Fig. 6. An example in sequential presentation from top left to top right, and bottom left to bottom right. It shows the generated visual queries, the related original images and the retrieved top-5 images. (image order: 5, linguistic cue: engine).

having ‘noise’ and ‘once’ together (not even in the same dataset), the visual query with mixed two cases can emerge when given the cues ‘noise’ and ‘once’ together. This point is important if the amount of data is very large, because one text can have the visual concept each, which they can work as additive prototypes.

The result of the overall retrieval performance is summarized in Table 3. It is done by checking whether more than one original image is retrieved for each linguistic cue in text dictionary during the incremental learning. If the large portion of the corpus is sparse, unsupervised learning methods confront the difficulty of learning the specific information for the discrimination. To show general characteristics of performance, we may ignore the cases of low frequencies. As a result, then, we get higher accuracy to retrieve relevant images.

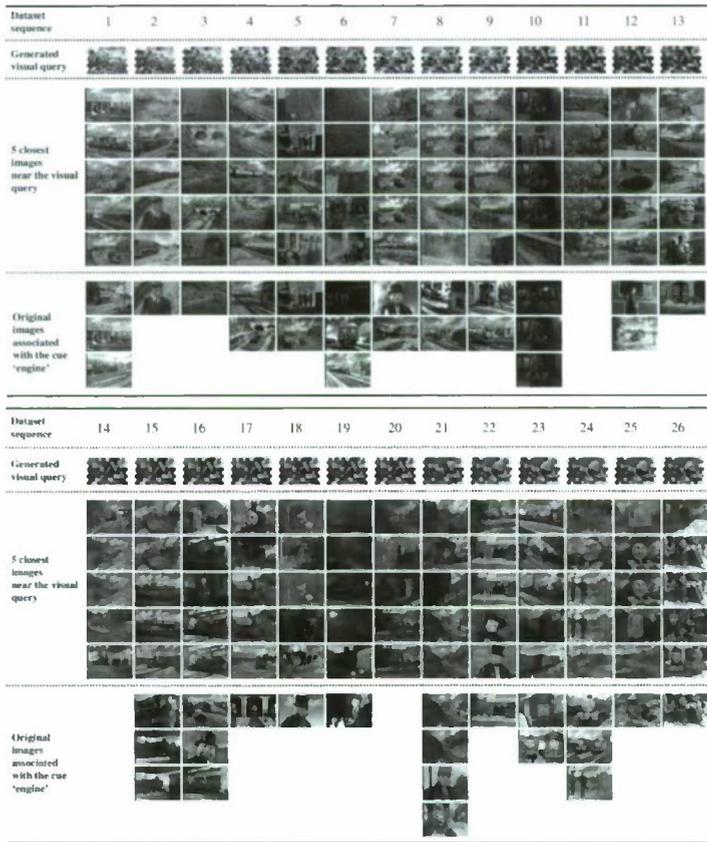


Fig. 7. An example in sequential presentation from top left to top right, and bottom left to bottom right. It shows the generated visual queries, the related original images and the retrieved top-5 images. (image order: 35, linguistic cue: engine).

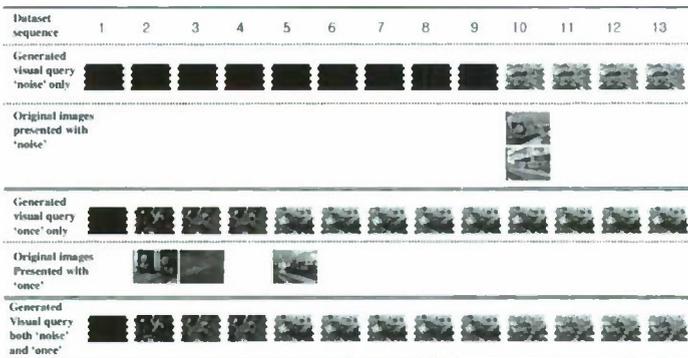


Fig. 8. An example of mixed words giving the cues 'noise' and 'once' by sequence 13. Even though they do not occur in the same instances, generated visual query reflects the original images together well. The reason for black visual queries in the left part comes from presenting no instance to learn 'noise' and 'once' yet.

Table 3. The overall performance of retrieval results in various tasks (order: 35)

Retrieval task		# of cases	Size of retrieved candidates (Top-N)			
			3	5	8	10
All cases	Successful cases	1256	334	462	617	692
	Percentage (%)		26.6%	36.8%	49.1%	55.1%
Cases of freq. ≥ 3	Successful cases	528	253	338	414	438
	Percentage (%)		47.9%	64.0%	78.4%	83.0%
Cases of freq. ≥ 5	Successful cases	380	215	286	336	343
	Percentage (%)		56.6%	75.3%	88.4%	90.3%
Cases of freq. ≥ 7	Successful cases	288	187	237	270	272
	Percentage (%)		64.9%	82.3%	93.8%	94.4%
Cases of freq. ≥ 10	Successful cases	208	154	190	203	204
	Percentage (%)		74.0%	91.4%	97.6%	98.1%

6 Concluding Remarks

We separated text-to-image retrieval task into two steps as follows: 1) text-to-image conversion and 2) image-to-image retrieval. And we proposed a method to generate visual query based on cross-modal associative learning by incremental hypernetwork models with the focus on the text-image conversion reflecting the related images from an image-text corpus. Experimental results show that the visual query generated by this method can be used for the image-to-image retrieval task. In this study, we just estimate with the small number of bases of the specific order (k -order hyperedges) without explicit learning process. We will go on to establish proper learning processes with unsupervised HNs and apply proper CBIR methods to the second step.

Acknowledgements

This work was supported in part by IT R&D Program of MKE/KEIT (K1002138, MARS), by NRF Grant of MEST (314-2008-1-D00377, Xtran), and by the BK21-IT program funded by Korean Government (MEST). The authors thank Jung-Woo Ha and Byoung-Hee Kim for helpful discussion. And we also thank Minsu Cho and Prof. Kyoung Mu Lee for image processing.

References

- [1] Datta, R., Joshi, D., Li, J., Wang, J.Z.: Image retrieval: Ideas, influences, and trends of the new age. *ACM Computing Surveys (CSUR)*, Article 5, 40(2) (2008)
- [2] The Stanford Encyclopedia of Philosophy, <http://plato.stanford.edu>
- [3] Tsybmal, A.: The problem of concept drift: definitions and related work, Technical Report TCD-CS-2004-15, Department of Computer Science, Trinity College Dublin, Ireland (2004), <http://www.cs.tcd.ie/publications/tech-reports/reports.04/TCD-CS-2004-15.pdf>

- [4] Ha, J.-W., Kim, B.-H., Kim, H.-W., Yoon, W.C., Eom, J.-H., Zhang, B.-T.: Text-to-image cross-modal retrieval of magazine articles based on higher-order pattern recall by hypernetworks. In: *The 10th Int. Symposium on Advanced Intelligent Systems (ISIS 2009)*, pp. 274–277 (2009)
- [5] Zhang, B.-T.: Hypernetworks: A molecular evolutionary architecture for cognitive learning and memory. *IEEE Computational Intelligence Magazine* 3(3), 49–63 (2008)
- [6] Jeon, J., Lavrenko, V., Manmatha, R.: Automatic image annotation and retrieval using cross-media relevance models. In: *The 26th Int. ACM SIGIR Conf. on Research and Development in Information Retrieval*, pp. 119–126 (2003)
- [7] Pan, J.-Y., Yang, H.-J., Faloutsos, C., Duygulu, P.: Automatic Multimedia Cross-modal Correlation Discovery. In: *The 10th ACM SIGKDD Conf. on Knowledge discovery and data mining*, pp. 653–658. Association for Computing Machinery, New York (2004)
- [8] Snoek, C.G.M., Worring, M.: Concept-based video retrieval. *Foundations and Trends in Information Retrieval* 2(4), 215–322 (2009)
- [9] Yan, R., Hauptmann, A.G.: A review of text and image retrieval approaches for broadcast news video. *Information Retrieval* 10(4-5), 445–484 (2007)
- [10] Li, D., Dimitrova, N., Li, M., Sethi, K.: Multimedia content processing through cross-modal association. In: *Proc. of the 11th ACM Int. Conf. on Multimedia (MM 2003)*, pp. 604–611 (2003)
- [11] van de Sande, K.E.A., Gevers, T., Snoek, C.G.M.: Evaluating Color Descriptors for Object and Scene Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2010) (in Press)
- [12] Chum, O., Philbin, J., Sivic, J., Isard, M., Zisserman, A.: Total Recall: Automatic Query Expansion with a Generative Feature Model for Object Retrieval. In: *Proc. 11th Int. Conf. on Computer Vision (ICCV 2007)*, pp. 1–8 (2007)
- [13] Joly, A., Buisson, O.: Logo Retrieval with a Contrario Visual Query Expansion. In: *Proc. 7th ACM Int. Conf. on Multimedia*, pp. 581–584 (2009)
- [14] Jiang, Y.-G., Ngo, C.-W.: Bag-of-Visual-Words Expansion using Visual Relatedness for Video Indexing. In: *Proc. 31st Int. ACM SIGIR Conf. on Research and Development in Information Retrieval*, pp. 769–770 (2008)
- [15] Feng, S., Manmatha, R., Lavrenko, V.: Multiple Bernoulli Relevance Models for Image and Video Annotation. In: *CVPR 2004 (2)*, pp. 1002–1009 (2004)
- [16] Block, N.: *Imagery*. MIT Press, Cambridge (1981)
- [17] Glasgow, J., Papadias, D.: Computational imagery. *Cognitive Science* 16, 355–394 (1992)
- [18] Roy, D., Hsiao, K.-Y., Mavridis, N.: Mental Imagery for a Conversational Robot. *IEEE Transactions on Systems, Man, and Cybernetics, Part B* 34, 1374–1383 (2004)

Sampling Bias in Estimation of Distribution Algorithms for Genetic Programming Using Prototype Trees

Kangil Kim, Bob (R.I.) McKay, and Dharani Punithan

Structural Complexity Laboratory, Seoul National University, Korea
<https://sc.snu.ac.kr>

Abstract. Probabilistic models are widely used in evolutionary and related algorithms. In Genetic Programming (GP), the Probabilistic Prototype Tree (PPT) is often used as a model representation. Drift due to sampling bias is a widely recognised problem, and may be serious, particularly in dependent probability models. While this has been closely studied in independent probability models, and more recently in probabilistic dependency models, it has received little attention in systems with strict dependence between probabilistic variables such as arise in PPT representation. Here, we investigate this issue, and present results suggesting that the drift effect in such models may be particularly severe – so severe as to cast doubt on their scalability. We present a preliminary analysis through a factor representation of the joint probability distribution. We suggest future directions for research aiming to overcome this problem.

1 Introduction

A wide range of evolutionary algorithms learn explicit probability models, sampling individuals from them, using the fitness of individuals to update the model. They range from Colomni and Dorigo's Ant Colony Optimization (ACO) [1] and Baluja's Population Based Incremental Learning (PBIL) [2] through Muehlenbein and Manig's Factorized Distribution Algorithm (FDA) [3] or Pelikan's Bayesian Optimization Algorithm (BOA) [4] to Salustowicz and Schmidhuber's Probabilistic Incremental Program Evolution (PIPE) [5]. Historically, different strands of this research have developed in relative isolation, and there is no acknowledged single term to describe them. In this paper, we refer to such algorithms as Estimation of Distribution Algorithms (EDAs), acknowledging that this may be wider-than-normal usage.

When EDAs are applied to Genetic Programming (GP) [6] problems, the most obvious question is what statistical model to use to represent the GP solution space, and how to learn it. This question has drawn most of the attention of researchers in this field, with consequent neglect of the sampling stage of EDA-GP algorithms.

In GP, many EDAs have used variants of the Probabilistic Prototype Tree (PPT) as their probability model, beginning with PIPE [5] and extending to Yanai and Iba's EDP [7], Sastry et al.'s ECG [8], Hasegawa and Iba's POLE [9], Looks et al.'s BOAP [10] and Roux and Fontupt's Ant Programming [11]. The PPT is a convenient model for representing probability distributions estimated from tree individuals. However Hasegawa and Iba already noted that it suffers from some representational problems,

and proposed the Extended Parse Tree (EPT) variant [9]. What has not been studied is the effect on sampling drift of its implicit dependence model.

Sampling drift effect is an important problem for all probability models. However the strict probability dependence in the PPT greatly amplifies this effect relative to the other major sources of bias in EDAs (selection pressure and learning bias), thus becoming a critical issue in scaling of PPT-based EDAs to large-scale problems.

In this paper, we examine this problem both empirically and mathematically. We designed two simple problems, closely related to the well-known one-max and max problems, with simple fitness landscapes to reduce the effects of other factors. We compare the behaviour of a PIPE model with a PBIL-style independent model to illustrate the amplified effect of sampling bias. We mathematically investigate how the factorised distribution implicit in the PPT model causes this increased sampling bias.

In section 2, we present a brief overview of EDAs and of PPTs. The experiments are described in section 3, with their results following in section 4. Section 5 analyse the factorisation implicit in the PPT. We discuss the implications of these results in section 6, drawing conclusions and proposing future directions in section 7.

2 Background Knowledge

2.1 Estimation of Distribution Algorithms

EDAs are evolutionary algorithms incorporating stochastic models. They use the key evolutionary concepts of iterated stochastic operations as shown below:

```

generate N individuals randomly
while not termination condition do
    Evaluate individuals using fitness function
    Select best individuals
    Construct stochastic model from selected individuals
    Sample new population from model distribution
end while

```

They differ from a typical evolutionary algorithm only in model construction and sampling. All EDAs use some class \mathcal{M} of probability models, and a corresponding decomposition of the structure of individuals. Model construction specifies a model from \mathcal{M} for each component. Sampling a new individual traverses the components, sampling a value from each model, so that the sample component distribution reflects the model's. In the simplest version, PBIL, the probability model is a vector of independent probability tables, one for each location of the phenotype.

2.2 Probabilistic Prototype Trees and EDAs

PPT-based EDAs use a tree structure to store the probability distribution. Given a pre-defined instruction set of maximum arity n , the PPT is an n -ary full tree storing a probability table over the set of instructions. PPT was first used in PIPE [5], where each node contained an independent probability table. ECGP [8] extended this by modelling dependence between PPT nodes as in the Extended Compact Genetic Algorithm [12].

EDP [7] instead conditioned each node on its parent. BOAP [10] learnt Bayesian networks (BN) of dependences in the PPT, while POLE [9] learnt BNs representing dependences in an "Extended Parse Tree", a variant of the PPT.

2.3 Benchmark Problems

One Max is the near-trivial problem of finding a fixed-length binary string maximising the sum of all bits [13]. Its fitness landscape is smooth with no local optima. Thus it is well-suited to the PBIL independent-probability model, using a probability vector $V = [E_1, \dots, E_n]$ over the value set $\{0, 1\}$ to represent the locations in the string.

The Max Problem is a generalisation of one-max, where the goal is to find the largest-valued tree that can be constructed from a given function set I and terminal set T , in a given depth D [14]. Typically $I = \{\times, +\}$ and $T = \{0.5\}$. This appears well-suited to the "independent" probability model of PIPE, in that each node of the PPT – in this case, a full binary tree – holds an independent probability table, giving the probability of selecting each element of $I \cup T$. The simplest case of max, $I = \{+\}$, $T = \{0, 1\}$ is closely related to one-max, in that once the system has found a full binary shape, the remaining problem, of filling the leaves with 1, is essentially the one-max problem. We note that in making this comparison, we are, in effect, mapping the nodes of the PPT tree to corresponding locations in a PBIL chromosome.

2.4 Grammar Guided Genetic Programming

To set the context for this study, we compare the performance of GP on the same problems; we can't use a standard GP system for this, because it is unable to enforce the constraints of the one-max problem. For fair comparison, we use a Grammar Guided GP system (GGGP) [15].

3 Experimental Analysis

Our experiments illuminate sampling drift in PPT-based EDAs, comparing it with a well-understood model (PBIL). We need to specify four aspects:

1. the probability model structures
2. the fitness functions
3. the EDA algorithm
4. experimental parameters

To illustrate, we use the max problem, and a slight variant of one-max, with the same target as max (but a more one-max-like fitness function). We compare with a conventional GGGP approach to show the intrinsic simplicity of these problems. For economy of explanation, we describe the max problem first.

3.1 Model Structures

The Genotype Representation is a 15-long string $X = X_1, \dots, X_{15}$. This can be used in either of two ways: the string can be modelled through an independent, PBIL-style genotype, or it can be mapped to a binary PPT of depth 3 (which has 15 nodes).

In the *PBIL structure* each location contains an independent probability table with three possible values, +, × and 0.5. The table is used to generate sample values at each generation, then is updated to reflect the sample distribution of the selected individuals.

In the *PPT structure* each location contains an independent probability table over the values +, × and 0.5, but each (except the leaves) has two children, with the relationship:

$$\begin{aligned} \text{left child}(X_i) &= X_{i \times 2} \\ \text{right child}(X_i) &= X_{i \times 2 + 1} \end{aligned}$$

“Independence” in the latter case must be taken with a grain of salt. While the probability tables in the PBIL structure are independent, the PPT structure introduces a dependence: the descendants of a node holding the (terminal) value 0.5 are not sampled. This is the primary issue under consideration here.

3.2 Max Problem Fitness Function

Fitness is defined by the following equation:

$$\text{itFit}(X_i) = \begin{cases} \text{itFit}(\text{left child}(X_i)) \times \text{itFit}(\text{right child}(X_i)) & \text{if } X_i = \times, 1 \leq i \leq 7 \\ \text{itFit}(\text{left child}(X_i)) + \text{itFit}(\text{right child}(X_i)) & \text{if } X_i = +, 1 \leq i \leq 7 \\ 0.0 & \text{if } X_i = \times, 8 \leq i \leq 15 \\ 0.0 & \text{if } X_i = +, 8 \leq i \leq 15 \\ 0.5 & \text{if } X_i = 0.5 \end{cases}$$

When +, × were used in leaf nodes, there is a problem in allocating fitness, since they have no children. To overcome this, in this case we give them fitness 0. The maximum value of this function (the target) corresponds to a full binary tree with + in the bottom two layers, and + or × in the top layer.

3.3 Variant One-Max Problem Fitness Function

The task is to find a string having a specific value in each location, defined by dividing the locations into three groups, as in equations 1.

$$\begin{aligned} L_1 &= \{X_1\} \\ L_2 &= \{X_i\} \quad 2 \leq i \leq 7 \\ L_3 &= \{X_i\} \quad 8 \leq i \leq 15 \end{aligned} \tag{1}$$

In this case, the fitness function is given by equation 2:

$$\text{omFit}(X) = \sum_{i=1}^{15} \text{loeFit}(X_i) \tag{2}$$

where

$$\text{loeFit}(X_i) = \begin{cases} 1 & \text{if } X_i = \times \quad \text{and } X_i \in L_1 \\ 1 & \text{if } X_i = + \quad \text{and } X_i \in L_2 \\ 1 & \text{if } X_i = 0.5 \quad \text{and } X_i \in L_3 \\ 0 & \text{else} \end{cases}$$

This differs from the typical one-max problem in two ways: there are three possible values, not two, and target values at differ with location. However neither makes much difference to the fitness landscape, which remains smooth, with no local optima.

3.4 EDA System

In these comparisons, we use a very simple EDA system so that the implications of the experiments are clear. In detail:

Selection: truncation. Given a selection ratio λ , the top λ proportion of individuals are selected. We varied the selection ratio λ to investigate the effect and scale of drift.

Model Update: the model structure was fixed for the whole evolution. Maximum likelihood was used to estimate the probabilities from the selected sample.

Sampling: we used Probabilistic Logic Sampling [16], the most straightforward sampling method, used in most EDA-GP systems.

To simplify understanding, two common EDA mechanisms which can slow drift, elitism and mutation, were omitted from the system

3.5 Parameter Settings

We used truncation selection with selection ratios ranging from 10% to 100% at a 10% interval. The population size was 100, and the algorithm was run for 200 generations. Each setting was run 30 times. Detailed parameters settings for the GGGP and EDA-GP runs are shown in table 1, while the grammar used for GGGP (with starting symbol EXP_1) is shown in table 2.

Table 1. Experimental Parameter Settings

General Parameters	Value	EDA Parameters	Value	GGGP Parameters	Value
Genotype Length Values	15 +, ×, 0.5	Operators Selection Ratios Update Sampling	Truncation 0.1, ..., 1.0 Max. Likelihood PLS	Operators Selection Size Cross. prob. Mut. prob.	Tournament 5 0.5 0.75
Population Generations Runs	50 200 30	Dependence PBIL PPT	independent PPT	Reproduction	Generational

Table 2. GGGP Grammar

$$\begin{aligned}
 EXP_i &\rightarrow EXP_{i+1} OP EXP_{i+1} & (0 < i < 4) \\
 EXP_4 &\rightarrow OP \\
 OP &\rightarrow + | \times | 0.5
 \end{aligned}$$

4 Result of Preliminary Experiments

4.1 One-Max Results

Figure 1 shows the performance of the two probability models, at various levels of selection. Each plot shows a particular structure for a range of different selection ratios.

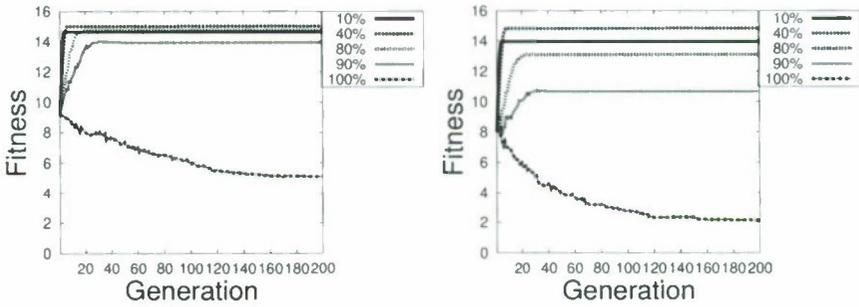


Fig. 1. Best Fitness vs Generation for One-max Variant (Structure : left, *PBIL*, right, *PPT*; percentage is the selection ratio)

Each line represents the best fitness achieved in each generation, for a particular selection ratio. By comparison, GGGP finds perfect solutions in 14.3 ± 4.9 generations.

We note that even for this near-trivial fitness function, *PPT* shows worse performance than *PBIL*. In the left-band plot, the *PBIL* structure finds a solution close to the optimum (15) at most selection ratios other than 90% and 100% (i.e. no selection). These results are replicated for the selection ratios not plotted, most showing performance very close to the optimum, as with the 40% selection ratio. By comparison, the *PPT* model shows much worse performance. In all selection ratios, *PPT* converges to sub-optimal solutions. The difference increases with weaker selection, with the 100% ratio showing a substantial decrease in fitness, below that achieved by random sampling. With selection pressure turned off, this drift is the result purely of sampling. With increasing selectivity, the drift effect becomes weaker, but still acts counter to the selection pressure.

4.2 Max Problem

This problem is much tougher than the previous. GGGP finds perfect solutions in 17.8 ± 8.0 generations. However EDA performance fares far worse. The *PBIL* model is unable to find the optimum solution (4) at any selection ratio, and the differences from the optimum are larger than for one-max. Given that the fitness function has epistasis, which *PBIL* is unable to model, this is not surprising. What is surprising is the even poorer performance of the *PPT* model. *PPT* appears well-matched to the fitness function, yet performs much worse than the naive *PBIL* model. *PBIL* is able to achieve fitnesses, for some selection ratios, of around 3.4, whereas *PPT* never exceeds 2.7. The effects are particularly marked around selection ratios from 10% through to 60%, with the differences becoming weaker by 80% to 90%, and essentially disappearing at a 100% selection ratio.

4.3 Performance of PPT

Overall, we see poor performance from the *PPT* model for both simple and complex problems. Even for the max problem – the kind of problem that *PPT* was designed to solve – it shows much worse performance than *PBIL*. The behaviour under 100% selection – i.e. pure sampling drift – suggests a possible cause: that sampling drift [17]

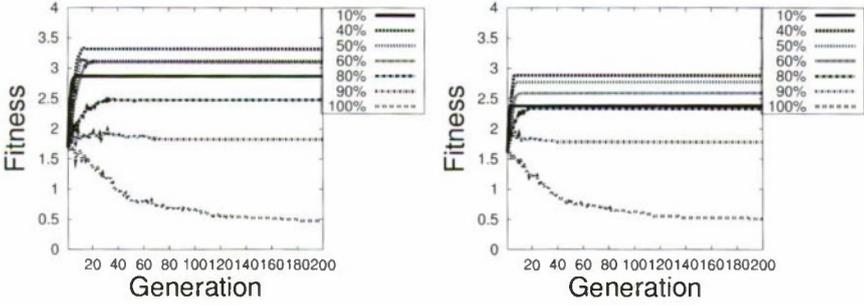


Fig. 2. Best Fitness vs Generation for Max (Structure : left, *PBIL*, right, *PPT*, percentage is the selection ratio)

may be the major influence on performance. The poor performance on the trivial fitness landscape of the onc-max variant supports this. The good performance of GGGP emphasizes just how damaging this effect is.

5 Analysis of the PPT Model

5.1 The Effects of Arity

In a PPT, each node represents a random variable which can take any of the possible instructions as its value.¹ Table 3 shows a typical example for the case of symbolic regression, with a function set consisting of the four binary arithmetic operators, four unary trigonometric and exponential operators, and a variable and constant, of arity 0.

Table 3. PPT Table for Symbolic Regression, Showing Arities

Instruction	Arity	Probability	Instruction	Arity	Probability
+	2	0.1	sin	1	0.1
×	2	0.1	cos	1	0.1
-	2	0.1	log	1	0.1
/	2	0.1	exp	1	0.1
<i>x</i>	0	0.1	<i>C</i>	0	0.1

The combining of nodes of different arities in the PPT model creates a dependence relationship between parent and child nodes, even though their probability distributions appear to be separate. If a node n_1 is sampled as sin, one of the child nodes – conventionally n_3 – loses the opportunity to sample an instruction. Therefore the probability of sampling n_3 is different from that of n_2 , the other child node. Thus although the probability distribution of n_3 is independent of the condition set of n_1 , n_3 is nevertheless

¹ Nodes at the maximum depth are only permitted values of zero arity, but for the sake of simplicity we omit this from consideration here.

dependent on the complete condition set of n_1 , because the probability of sampling an instruction for n_3 is 0 in the case where a unary function or variable is sampled at n_1 .

To clarify this dependency, we transform the PPT probability distribution to a semi-degenerate Bayesian network.²

5.2 Conversion to Semi-degenerate Bayesian Network

Undefined Instruction. In the PPT, each node's probability table cannot be directly treated as a random variable, because the probability distribution for some conditions of the parent is not recorded in the table. To cover this case, where a node can not select any value, we define an additional value U , for 'undefined value'. Taking a simple case with just three values, $+$, \sin and C , an independent PPT might have probabilities of 0.4 for $+$ and \sin , and 0.2 for C . Taking account of the parent-child dependencies, we could represent the overall conditional dependency of a random variable for a node given its parent, as in figure 3. In the parent node of M_4 , any of $+$, \sin , C or U might be sampled. When C , constant, is sampled, M_4 is not able to sample any value, so that the probabilities for selecting $+$, \sin and C are zero; to represent that no instruction can be sampled in this condition, we allocate the 'undefined' instruction a probability of 1.0. If the parent node is sampled as 'undefined', M_4 must also be undefined.

M_4

	$+$	\sin	C	U
$+$	0.4	0.4	0.0	0.0
\sin	0.4	0.4	0.0	0.0
C	0.2	0.2	0.0	0.0
U	0.0	0.0	1.0	1.0

Fig. 3. Transformed Probability table of PPT

Figure 4 shows more detail, illustrating how a simple three-node PPT can be transformed into a (semi-degenerate) BN. Note that the probability structures of the left and right children differ (because of the differing effects of the \sin function in the parent).

5.3 Factorization of Full Joint Distribution

Dependent Variable. In the resulting BN, the transformed nodes become conditionally dependent on their parent nodes (there are only two exceptions – either the node is always undefined, hence unreachable and may be omitted from the PPT, or else the

² In standard terminology, tables without zeros are said to be non-degenerate, and tables containing only 0.0 and 1.0 are degenerate. We introduce the term 'semi-degenerate' for the intermediate case, of tables containing 0.0 but not necessarily 1.0.

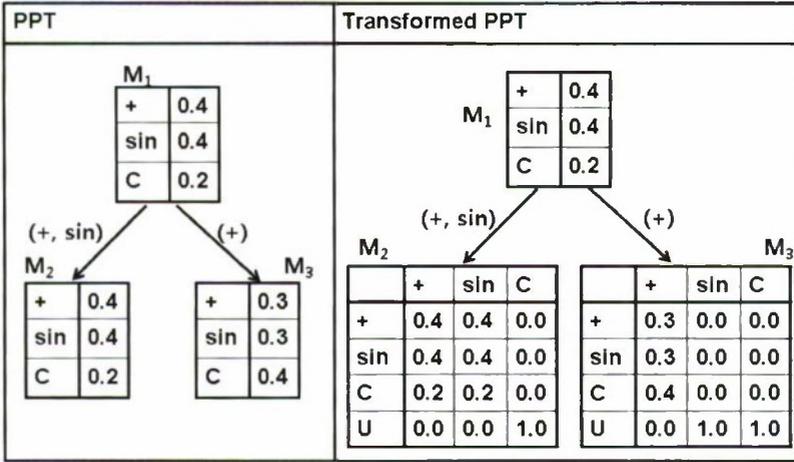


Fig. 4. Transformation from PPT to semi-degenerate BN

node is always defined, implying that the parent node cannot sample a terminal, an unreasonable situation in GP – both may be safely ignored).

In the simplest PPT case, where each node’s value is assumed probabilistically independent of the other nodes, the only dependence is that arising above. That is, this simple case corresponds to the assumption that each node is conditionally independent of all other nodes in the PPT, conditioned only on its parents. Thus the probability distribution of node x can be represented by $p(x|parent\ of\ x)$, and the full joint probability distribution of the transformed PPT as:

$$p(X) = \prod_i p(x_i|x_{parent\ of\ i}) \tag{3}$$

Of course, more complex dependencies between PPT nodes may give rise to more complex dependencies in the corresponding BN, but the dependence of the child on its parents will always remain.

Sampling Bias. This factorization of the joint distribution gives us a way of understanding the rapid diversity loss in PPT-based EDAs. In PLS sampling, for each random variable, the sample size is the same in the transformed PPT. However the actually meaningful instructions exclude undefined instructions. The size of the sample actually used to generate meaningful instructions reduces (exponentially) with depth. This is the cause of the rapid diversity loss due to sampling drift: unlike other EDAs, in which the sample size is the same across all variables, drift increases due to reduced sample size with depth. Figure 5, shows the population (phenotype) entropy at each generation. We only show the 100% selection ratio, because there, there is no diversity loss due to selection, the whole loss is the result of sampling drift. In both problems, the loss of diversity due to sampling drift is much greater in the PPT representation than in the PBIL representation.

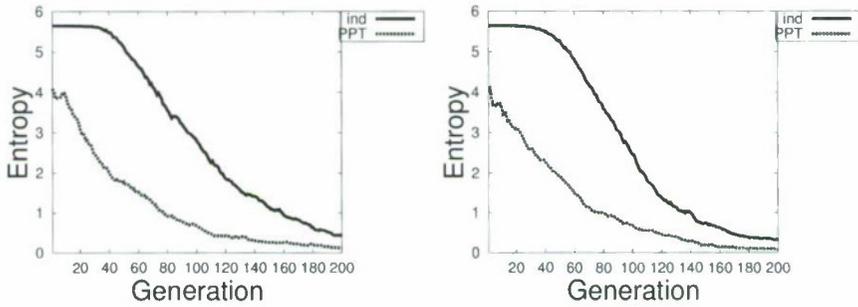


Fig. 5. Entropy of Population vs Generation (Left: One-max Variant; Right: Max (ind : independent – PBIL – structure))

6 Discussion

The importance of these results lie not merely in their direct implications for this trivial problem, but in their implications for PPT-based EDAs for GP. Compare these problems with typical GP problems. The dependency depth is atypically small, corresponding to a GP tree depth bound of only 3. The dependency branching is typical, or even slightly below average, for GP. And of course, the fitness landscape is vastly simpler than most GP problem domains. If this is so, why has EDA-GP been able to succeed, and even demonstrate good performance on some typical GP problems? We believe it is due to masking of the problem of accelerated drift under sampling in typical implementations.

These implementations generally incorporate mechanisms reducing the effect of sampling drift: better selection strategies and model update mechanisms, adding elitism and mutation all contribute to this reduction. In addition, our problem is tougher than typical GP problems in one respect: there is only one solution (two for the max problem). Most problem domains explored by GP have symmetries, so that eliminating a solution may not stymie exploration. Thus EDA-GP has been able to work well for GP test problems. However the drift effect worsens exponentially with tree depth, while these ameliorating mechanisms only scale linearly. Perhaps this is why EDA-GP has so far been limited to demonstrations on test problems rather than practical applications.

Some previous PPT research, notably Hasegawa and Iba's POLE [9], incorporates measures to ameliorate sampling drift using the Extended Parse Tree. Here, our focus is to clarify the effect of accelerated drift due to PPT dependency, as a preliminary to investigating solutions.

7 Conclusions

Diversity loss due to sampling is a well-known problem in EDA research, and has been carefully studied for independent probability models. It is well-known that the the problem worsens in probabilistic dependency models, and some lower bounds for the effect have already been found [17]. However there does not appear to have been previous publication of the effects on PPT-based (branching) EDAs.

By studying the sampling drift effect of two structures, on a near-trivial optimisation problem and another only slightly harder, we were able to see the importance of this diversity loss. The effects are sufficient to cast doubt on the scalability of most current approaches to EDA-GP. Can these problems be overcome? Can scalable EDA-GP systems be built? We believe it to be possible, but not easy. Any remedy must counteract the depth dependence of the drift. This probably eliminates variants of some of the traditional methods. For example, it is difficult to see how to incorporate dependence depth into population-based mechanisms such as elitism. Similarly, it doesn't seem easy to use mutation or similar mechanisms in a useful depth-dependent way. On the other hand, it may be possible to incorporate depth-based mechanisms into model update and/or sampling in ways that might be able to overcome the depth-dependence of sampling drift, and so permit scaling.

In the near future, we plan to extend this work in three directions. The first, already in progress, involves experimental measurement of diversity loss to gauge the extent of acceleration of the sampling drift effect. The second, in prospect, will attempt to mathematically estimate the diversity loss through sample size estimation. The third extends this work to grammar-based GP EDA systems (i.e. those not based on PPTs). Similar problems of accelerated sampling bias occur in these systems, though it is more difficult to isolate clear demonstrations of this.

Acknowledgment

We would like to thank Pedro Larranāga, Concha Bielza and Xuan Hoai Nguyen for their insightful involvement in discussions that led to this work. This work was supported by the Brain Korea 21 Project. The ICT at Seoul National University provided research facilities for this study.

References

1. Colomi, A., Dorigo, M., Maniezzo, V., et al.: Distributed Optimization by Ant Colonies. In: Varela, F.J., Bourgine, P. (eds.) *Toward a Practice of Autonomous Systems: Proceedings of the First European Conference on Artificial Life*, pp. 134–142. MIT Press, Cambridge (1991)
2. Baluja, S.: Population-based incremental learning: A method for integrating genetic searching based function optimization. Technical Report CMU-CS-94-163, Computer Science Dept., Carnegie Mellon University, Pittsburgh, PA, USA (1994)
3. Mühlenbein, H., Mahnig, T.: The factorized distribution algorithm for additively decomposed functions. In: *Proceedings of the 1999 Congress on Evolutionary Computation*, pp. 752–759. IEEE Press, Los Alamitos (1999)
4. Pelikan, M., Goldberg, D., Cantu-Paz, E.: BOA: The Bayesian optimization algorithm. In: *Proceedings of the Genetic and Evolutionary Computation Conference, GECCO 1999*, vol. 1, pp. 525–532 (1999)
5. Salustowicz, R., Schmidhuber, J.: Probabilistic incremental program evolution. *Evolutionary Computation* 5(2), 123–141 (1997)
6. Koza, J.: *Genetic programming: on the programming of computers by means of natural selection*. The MIT Press, Cambridge (1992)

7. Yanai, K., Iba, H.: Estimation of distribution programming based on bayesian network. In: Proceedings of the Congress on Evolutionary Computation, Canberra, Australia, pp. 1618–1625 (December 2003)
8. Sastry, K., Goldberg, D.E.: Probabilistic model building and competent genetic programming. In: Riolo, R.L., Worzel, B. (eds.) Genetic Programming Theory and Practise, pp. 205–220. Kluwer, Dordrecht (2003)
9. Hasegawa, Y., Iba, H.: A bayesian network approach to program generation. IEEE Transactions on Evolutionary Computation 12(6), 750–764 (2008)
10. Looks, M., Goertzel, B., Pennachin, C.: Learning computer programs with the bayesian optimization algorithm. In: GECCO 2005: Proceedings of the 2005 Conference on Genetic and Evolutionary Computation, pp. 747–748. ACM, New York (2005)
11. Roux, O., Fonlupt, C.: Ant programming: or how to use ants for automatic programming. In: Proceedings of the Second International Conference on Ant Algorithms (ANTS 2000), Belgium (2000)
12. Harik, G., Lobo, F., Sastry, K.: Linkage Learning via Probabilistic Modeling in the Extended Compact Genetic Algorithm (ECGA). In: Scalable Optimization via Probabilistic Modeling, vol. 33, pp. 39–61. Springer, Heidelberg (2006)
13. Schaffer, J., Eshelman, L., Offutt, D.: Spurious correlations and premature convergence in genetic algorithms. Foundations of Genetic Algorithms, pp. 102–112 (1991)
14. Gathercole, C., Ross, P.: An adverse interaction between crossover and restricted tree depth in genetic programming. In: GECCO 1996: Proceedings of the First Annual Conference on Genetic Programming, pp. 291–296. MIT Press, Cambridge (1996)
15. Whigham, P.A.: Grammatically-based genetic programming. In: Rosca, J. (ed.) Proceedings of the Workshop on Genetic Programming: From Theory to Real-World Applications, pp. 33–41 (1995)
16. Henrion, M.: Propagating uncertainty in bayesian networks by probabilistic logic sampling. In: Uncertainty in Artificial Intelligence 2 (UAI 1986), pp. 149–163. North Holland, Amsterdam (1986)
17. Shapiro, J.L.: Diversity loss in general estimation of distribution algorithms. In: Runarsson, T.P., Beyer, H.-G., Burke, E.K., Merlo-Guervós, J.J., Whitley, L.D., Yao, X. (eds.) PPSN 2006. LNCS, vol. 4193, pp. 92–101. Springer, Heidelberg (2006)

Identification of Non-referential Zero Pronouns for Korean-English Machine Translation

Kye-Sung Kim, Seong-Bae Park, Hyun-Je Song, Se Young Park, and Sang-Jo Lee*

Department of Computer Engineering
Kyungpook National University
702-701 Daegu, Korea

{kskim, sbpark, hjsong, sypark}@sejong.knu.ac.kr, sjlee@knu.ac.kr

Abstract. The common use of null arguments is one of the most critical issues in pro-drop languages. When translating Korean into other languages, the omitted elements should be replaced with appropriate pronouns to get grammatical target sentences. One of the most important issues when dealing with zero pronouns is to determine the referentiality of zero pronouns. Since, like expletive 'it' in English, omitted elements do not have always explicit referents, it is important to determine whether a zero pronoun is referential or not. In this paper, we focus on identifying non-referential zero pronouns. Since non-referential zero pronouns are likely to occur in similar contexts, referentiality determination in this paper is regarded as the identification of clauses containing non-referential zero pronouns. Our method outperforms the baseline systems using n-grams and bag of words, and achieves the F-measure of 0.51 and 0.78.

Keywords: zero pronoun, ellipsis, referentiality, anaphoricity, parse tree kernel.

1 Introduction

In pro-drop languages such as Chinese, Japanese and Korean, it is important to identify referents of missing elements which frequently occur in sentences. These omitted elements are often called zero pronouns, and the resolution of zero pronouns is of importance for various applications in natural language processing such as machine translation, text summarization, information extraction, and so on.

Zero pronouns are divided into three groups according to the positions in which the referents are understood: anaphora, cataphora and exophora [1]. That is, all zero pronouns do not have explicit referents in sentences. For that reason, recent work related to reference resolution has attempted to determine the referentiality (or anaphoricity) of nominal expressions including zero pronouns [11,12,17]. In the context of zero pronoun resolution, referentiality determination is the task of judging whether a given zero pronoun is referential or non-referential. If its explicit referent (or antecedent) is found in the text, the zero pronoun is classified as referential (or anaphoric); otherwise, it is classified as non-referential (or non-anaphoric). However, the performance of referentiality determination for zero pronouns is not satisfactory enough, because it is difficult

* Corresponding author.

(1)	철수-가 Cheolsu-NOM	내기-에 bet-COMP	지-니까 lose-PRED	(ϕ_1)-가 (ϕ_1)-NOM	싱술을 get cross-PRED	부린다 PUNC	
When Cheolsu loses a bet, _____ [get] cross.							
(2)	(ϕ_2)-이 (ϕ_2)-NOM	2시경-이 be about two o'clock-PRED	태양-이 sun-NOM	산허리 아래-로 behind the mountain-LOC	내려간다 set-PRED	PUNC	
If _____ [be] about two o'clock, the sun sets behind the mountain.							
(3)	스쿠아-가 skua-NOM	해면-에 the sea level-LOC	가까이 close-MOD	있-길래 be flying-PRED	(ϕ_3)-가 (ϕ_3)-NOM	유심히 carefully-MOD	살펴보-았-더니 watch-PAST-PRED
(ϕ_4)-기 (ϕ_4)-NOM	푸른 눈 blue-eyed-MOD	코모란트-를 cormorant-OBJ	공격하-고 있-었다 be attacking-PAST-PRED	PUNC			
A skua is flying close to the sea level, so _____ [be] carefully watched, and _____ [he] attacking a blue-eyed cormorant.							

Fig. 1. An example of sentences with zero pronouns

to distinguish non-referential from referential uses of the same forms. Most of previous studies have regarded all cases which fail to identify the referents of zero pronouns as non-referential. However, this is not an appropriate solution for the referentiality of zero pronouns, since there can be errors in referent identification for zero pronouns.

Figure 1 shows an example of sentences containing zero pronouns. In Figure 1, zero pronouns ϕ_1 and ϕ_4 are referring to 'Choelsu' and 'skua' in the same sentence respectively. However, the referents of ϕ_2 and ϕ_3 do not appear in the text. Thus, ϕ_2 is the zero pronoun that refers to time, and ϕ_3 is referring to the speaker which is a discourse participant. In the translation of Korean to English, non-referential zero pronoun ϕ_2 should be translated into 'it', and ϕ_3 should be replaced with 'i' (or 'we'). However, it is difficult to obtain additional information such as gender, number and person during translation, because the referents of non-referential ϕ_2 and ϕ_3 do not explicitly appear in sentences. Also, in the case of referential zero pronouns, such information is not always provided. Therefore, the referentiality of zero pronouns should be considered before translating, and has been considered as one of the most important issues to be addressed for practical applications like machine translation.

This paper proposes a method for identifying non-referential zero pronouns in sentences. Previous studies have determined non-anaphoric cases through pairwise comparisons between a zero pronoun and its antecedent candidates [11], and in most cases, do not learn non-anaphoric cases from non-anaphoric training examples. Thus, the referentiality of zero pronouns in previous work on evaluating the preference of antecedent candidates is determined by parametric models or by methods of identifying non-anaphoric cases indirectly [10,15,16]. In this paper, we attempt to identify non-referential cases directly from non-anaphoric training instances. Since they are likely to occur in similar contexts, the proposed model measures the syntactic similarity between the contexts in which zero pronouns occur. To do this, structural information of clauses is used for our experiments. In addition, the majority of zero pronouns occur in subject grammatical positions. The rate of subject drop is approximately 94% in Korean, which is significantly higher than zeros in other positions [12]. Therefore, this paper focuses on determining the referentiality of subject zero pronouns. Referentiality determination

in this paper is regarded as the identification of clauses containing non-referential subject zero pronouns. In our experiments, support vector machines with a parse tree kernel [6,13] are used to examine the structural similarity between clauses.

The remainder of this paper is organized as follows. Section 2 surveys previous work on zero pronouns. Section 3 proposes a method for identifying non-referential zero pronouns in machine learning approach and Section 4 presents experimental results and the conclusion is given in Section 5.

2 Related Work

Most studies on reference resolution including zero pronouns are widely divided into two groups. One is based on heuristic rules or theoretical approaches such as Centering theory [2]. Centering theory provides a model of local coherence in discourse, and has usually been used to resolve pronouns in English. However, it is difficult to deal with all types of zero pronouns in the framework of Centering, since it is not easy to identify the referentiality of zero pronouns in pro-drop languages which allow missing subjects such as Chinese, Japanese and Korean. Roh [14] proposed a cost-based centering model for generating zero pronouns corresponding to anaphoric expressions in order to produce a coherent text in Korean. However, there is a problem in applying this model directly to zero pronoun resolution. In addition, the use of non-referential zero pronouns is not considered in the revised centering model [14].

The other approach is based on machine learning methods [20]. Previous studies can be reclassified according to whether or not anaphoricity (or referentiality) determination is separated from antecedent identification. First, previous work focusing on antecedent identification classifies noun phrases intervening between a zero pronoun and its referent or noun phrases which are not involved in coreference chains as negative instances [7,10]. These studies have regarded zero pronouns which fail to identify their referents as non-referential cases. However, it is not reasonable to determine that zero pronouns in such cases are all non-anaphoric, since there can be errors in antecedent identification model. Recent studies have attempted to determine the anaphoricity of zero pronouns in a separated step [10,15,16]. For zero pronouns in Korean, Han [12] has attempted to identify the referents according to anaphoric and non-anaphoric uses of zero pronouns. However, the model proposed by Han [12] was designed based on morpho-syntactic information. In addition, by trying to characterize anaphoric and non-anaphoric cases in the similar manner, they did not provide evidence for anaphoricity determination. Iida [15] has presented anaphoricity determination model using syntactic patterns. The importance of structural information extracted from parse tree has been shown in recent work [15,18]. In Iida [15]'s work, they have proposed tournament-based model which learns the relative preference between candidates. The most likely candidate antecedent of a zero pronoun is selected through the tournament model, and the final antecedent is determined by determining whether the zero pronoun and the chosen candidate antecedent are anaphoric. However, their model of anaphoricity determination is parametric, and is built on the results of antecedent identification.

The concern of anaphoricity determination has also been expressed in English [8,17]. Recently, Bergsma [17] presented an approach to detecting non-referential pronouns

Type	Example
(1) Deictic	a. \emptyset 점심 먹어라. (You) eat lunch. b. 한국팀이 이겨서 \emptyset 기쁘다. (I) am happy that the Korean team won.
(2) General Situational	c. \emptyset 벌써 열시다. (It) is ten o'clock already. d. 영희는 온난화에 대해서 설명했다. Younghee explained <u>regarding</u> global warming.
(3) Indefinite Personal	e. \emptyset 호랑이를 잡으려면 \emptyset 산에 가야 한다. If (one) wishes to catch a tiger, (one/he) must go to the mountains.

Fig. 2. An example of non-referential zero pronouns

in text based on the distribution of the pronoun's context, in order to determine the referentiality of English pronoun 'it'. However, in pro-drop languages that allow free word order and frequent ellipsis of elements, the occurrence and referentiality of zero pronouns should be more carefully considered.

Thus, determining the referentiality in the use of referring expressions is of importance for reference resolution and many applications in natural language processing. The referentiality of zero pronouns has emerged as an important issue in pro-drop languages, but the performance of referentiality determination for zero pronouns is still not satisfactory.

3 Identification of Non-referential Zero Pronouns

3.1 Non-referential Zero Pronouns

Zero pronouns that do not have explicit antecedents in the same text are regarded as non-referential ones in this paper. From this view, exophoric zero pronouns [1] also are treated as non-referential although they refer to something extralinguistic. Therefore, in this paper, non-referential zero pronouns can be classified as follows [12].

- (1) Deictic zero pronoun
- (2) General situational zero pronoun
- (3) Indefinite personal zero pronoun

Figure 2 shows non-referential uses of zero pronouns. Zero pronouns which refer to discourse participants such as the speaker and the hearer are classified as deictic reference in type (1), and zero pronouns in type (2) refer to time, weather, general situation and so on. In addition to that, idiomatic expressions such as "regarding", "according to", "for the sake of" and so on are also classified into type (2), as done in Han [12]. A pronominal use of zero pronouns which refer to a generic person like "one" is found in type (3). Sometimes, indefinite zero pronouns in type (3) can be used to refer to specific entities which are not explicitly mentioned in the context. In this paper, three types of zero pronouns described above and zero pronouns with verbal or clause antecedents are considered as non-referential.

3.3 Support Vector Machine with Parse Tree Kernel

The identification of clauses with non-referential subject zero pronouns can be considered as a binary classification task. Let $D = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)\}$ be a set of training examples where $y_i \in \{-1, +1\}$ and $\mathbf{x}_i = c_i$. Here, each c_i is a clause and y_j is the class label associated with this training sample. The value $+1$ of y_i implies that there is a non-referential subject zero pronoun in clause c_i .

The identification of non-referential zero pronouns is to estimate a function $f : \mathbf{X} \rightarrow Y$. After the function f parameterized by θ is trained with D , the relationship detection y^* of an unlabeled example \mathbf{x} can be determined by

$$y^* = \arg \max_{y \in \{-1, +1\}} (f(\mathbf{x}, \theta) = y).$$

Since our task is a binary classification, support vector machines (SVM) are adopted as an implementation of the function f . The decision function of SVMs is defined by

$$y^* = \text{sgn}(\sum_{j \in SV} y_j \alpha_j \phi(\mathbf{x}_j) \cdot \phi(\mathbf{x}) + b), \tag{1}$$

where ϕ is a non-linear mapping function from \mathbb{R}^N to \mathbb{R}^H ($N \ll H$), SV is a set of support vectors, and $\alpha_j, b \in \mathbb{R}, \alpha_j \geq 0$. The mapping function ϕ should be designed such that all training examples are linearly separable in \mathbb{R}^H space.

Since it is crucial to design an explicit form of ϕ , the inner product of $\phi(\mathbf{x}_j)$ and $\phi(\mathbf{x})$ is computed using a simple kernel such that

$$K(\mathbf{x}_j, \mathbf{x}) = \phi(\mathbf{x}_j) \cdot \phi(\mathbf{x}).$$

As a result, when a kernel K_P is designed to compute the inner product between clauses, Equation (1) is rewritten as

$$y^* = \text{sgn}(\sum_{j \in SV} y_j \alpha_j K_P(\mathbf{x}_j, \mathbf{x}) + b). \tag{2}$$

In order to apply SVM to our task, a number of positive and negative examples used as D are generated.

A parse tree kernel is used to measure the syntactic similarity between clauses. The parse tree kernel is a specialized convolution kernel introduced by Hausler [3] and efficiently reflects structural information [6,13]. In the vector representation of a parse tree, the features correspond to the subtrees that can possibly appear in the parse tree. The value of a feature is the frequency of the corresponding subtree in the parse tree. The inner product of the vector representations of two trees, T_1 and T_2 is computed using the following equation [6].

$$\begin{aligned} &< V_{T_1}, V_{T_2} > \\ &= \sum_i \#st_i(T_1) \cdot \#st_i(T_2) \\ &= \sum_i (\sum_{n_1 \in N_{T_1}} I_{st_i}(n_1)) \cdot (\sum_{n_2 \in N_{T_2}} I_{st_i}(n_2)) \\ &= \sum_{n_1 \in N_{T_1}} \sum_{n_2 \in N_{T_2}} C(n_1, n_2) \end{aligned} \tag{3}$$

where $\#st_i(T)$ is the frequency of a subtree st_i in T , and N_{T_1} and N_{T_2} are the sets of nodes in T_1 and T_2 respectively. $I_{st_i}(n_1)$ is a function that returns the frequency of st_i rooted at n_1 in T_1 , and $C(n_1, n_2)$ is the sum of the product of the numbers of times each subtree appears at n_1 and n_2 .

4 Experimental Results and Analysis

4.1 Dataset

For our experiments, the parsed corpus which is a product of STEP 2000 project supported by Korean government is used. We first manually identified subject zero pronouns in the parsed corpus, and then the complex compound sentences with one or more subject zero pronouns were extracted from the parsed corpus. A simple statistics on the dataset is given in Table 1. The number of selected sentences is 5,221 and the sentences are segmented into 20,748 clauses (on average, 3.97 clauses/sentence and 7.67 words/clause).

Table 1. A simple statistics on the dataset used in our experiments

	Number
Sentences	5,221
Clauses	20,748
Clauses in which subject zero pronouns occur	13,171

Table 2 shows the distribution of subject zero pronouns observed from our dataset. In Korean, the large proportion of zero pronouns can be resolved in the same sentences in which they occur as shown in Table 2. However, the number of extra-sentential zero pronouns corresponding to non-referential is also not small. Extra-sentential ones in our dataset make up 76% of non-intrasentential ones. Therefore, it is important to distinguish non-referential ones from the sentences in which zero pronouns occur. Since there no exist their explicit referents within and between sentences, it will be effective to deal with the non-referential use of zero pronouns at the sentence level. This paper focuses on identifying non-referential zero pronouns in the context of clauses.

Table 2. The distribution of subject zero pronouns observed from our dataset

Intra-sentential	Inter-sentential	Extra-sentential
10,371	666	2,134
(78.74%)	(5.06%)	(16.20%)

4.2 Experimental Results and Analysis for Referentiality Determination

Our experiments are performed in five-fold cross validation and SVM_{light} [4] is used as classifiers. The accuracy and F-measure are used to evaluate the results of the identification of non-referential zero pronouns, and these are calculated as follows. The balanced

F-score which is the harmonic mean of recall and precision is used in our experiments. The results are shown in Table 3.

$$\text{Accuracy} = \frac{\text{number of correctly classified clauses}}{\text{total number of clauses}}$$

$$\text{Precision} = \frac{\text{number of correctly identified non-referential zero pronouns}}{\text{number of identified non-referential zero pronouns}}$$

$$\text{Recall} = \frac{\text{number of correctly identified non-referential zero pronouns}}{\text{number of true non-referential zero pronouns}}$$

Table 3. The performance of referentiality determination of subject zero pronouns

Model	Accuracy	F-measure
Voting	88.55%	–
N-grams (n=6)	90.06%	6.92
<i>BOW_{clause}</i>	92.12%	40.19
STRUC	92.12%	42.13
STRUC+ (r=0.8)	92.17%	51.09

To investigate the effect of structural information in this study, ‘n-grams’ and ‘BOW’ models are used as baseline systems. ‘N-grams’ model is based on the context surrounding zero pronouns regardless the division of clauses. In this paper, three words preceding and following zero pronouns are extracted as features of the ‘N-grams’ model. It can be viewed as a simplified version of the model introduced by Bergsma [17]. In ‘voting’ model, the final classification decision is taken by a simple majority vote. When the majority agree, it is classified as ‘positive’, and the accuracy is 88.55%. However, since this leads to the result that the identification of non-referential zero pronouns is not performed, further research is needed. In the bag of words (BOW) model, a clause is represented as unordered collection of words. As shown in Table 3, ‘BOW’ model based on clauses outperforms ‘n-grams’ model. In particular, the context size of ‘n-grams’ is similar to average length of clauses in ‘BOW’ model, but the recall of ‘n-grams’ is quite low. It implies that information obtained from the unit of clauses is useful in identifying non-referential zero pronouns. ‘STRUC’ and ‘STRUC+’ are models using structural information of clauses proposed in this paper. Here, ‘STRUC’ is using syntactic features obtained from the parse tree of clauses and ‘STRUC+’ model combines the syntactic features and a set of features extracted from words which occur in clauses, similarly to the BOW model. Thus, the composite kernel K for identifying non-referential zero pronouns is then

$$K = r \cdot K_1 + (1 - r) \cdot K_2,$$

where r ($0 \leq r \leq 1$) is a mixing parameter, and K_1 and K_2 are a parse tree kernel and a polynomial kernel with degree 3 respectively. In our experiments, the parameter r is set to 0.8 empirically, as shown in Figure 4. The fact that the performance with larger r is superior to that with small r implies that syntactic information is more positively related

to the identification of non-referential zero pronouns. Although this paper focuses on investigating the effect of structural information in the identification of non-referential zero pronouns, overall performance will be much better if a composite kernel using both structural information and semantic information is used in the future.

As shown previously, the performance of our method outperforms baseline systems. However, while the accuracy of the proposed models is quite high, the performance in terms of the f-measure is not satisfactory. This may be related to the problem of imbalanced data sets. In our dataset, the number of negative samples is much larger than that of positive ones and is approximately nine times higher than that of positive ones. A classifier induced from an imbalanced data set has, typically, a low error rate for the majority class and an unacceptable error rate for the minority class. In this situation, it is important to accurately classify the minority class in order to reduce the overall cost. In order to solve these problems, several methods can be considered such as reweighing, undersampling, and resampling [5,9]. In this paper, random under-sampling is considered, which involves under-sampling the majority class samples at random until their numbers matched the number of minority class samples. The results of sampling are shown in Table 4 and Figure 4. In our method, the precision and recall after sampling are 79.30% and 78.00% respectively. This shows that the problem of imbalanced data sets is significant in the identification of non-referential zero pronouns. In the future, this study will investigate the use of ensemble methods such as bagging and boosting to deal with imbalanced data.

Table 4. A performance comparison of sampling in the identification of non-referential zero pronouns using structural information ($r=0.8$)

	Accuracy	Precision	Recall	F-measure
Before sampling	92.17	87.45	39.71	51.09
After sampling	78.81	79.30	78.00	78.64

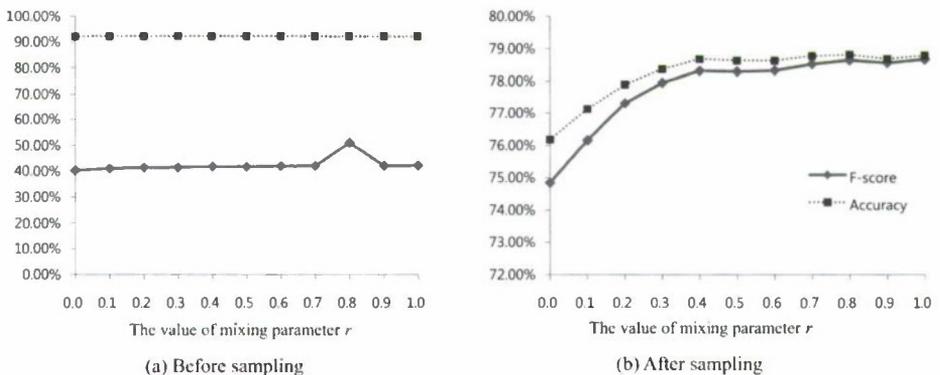


Fig. 4. A comparison of performance before and after sampling

4.3 An Application to Identification of Subject Shareness

Before applying our method to machine translation, this paper attempts to investigate the effect of the referentiality in zero pronoun resolution. Frequent omissions of subjects in Korean sentences imply that several predicates can share one subject. This is related to the subject-sharing problem of clauses [18]. When identifying antecedents of omitted subjects in intra-sentential resolution, it is necessary to determine whether their antecedents exist in the same sentences. In this situation, in order to investigate how referentiality determination affects subject shareness problem, this paper applies the referentiality determination to the model proposed by Kim [18]. Thus, if non-referential zero pronouns identified correctly by referentiality determination are excluded before antecedent identification, the performance of the identification of subject shareness may be improved.

Table 5. The effect of referentiality determination in subject shareness identification (SSI)

	Accuracy	Precision	Recall	F-measure
SSI	76.34	69.55	61.58	65.30
Referentiality+SSI	76.81	70.52	68.64	69.56

Table 5 shows the results of subject shareness, and these results indicate that the referentiality determination can play a positive role in the model of subject shareness. Therefore, it will be very useful for zero pronoun resolution or practical applications like machine translation if the performance of referentiality is more stable.

5 Conclusion

Referential expressions including zero pronouns commonly occur in texts. The identification of objects referred to by them is an important research area in natural language understanding. Like expletive ‘it’ or ‘there’ pronouns in English, zero pronouns do not always refer to objects which explicitly occur in texts. Therefore, it is important to distinguish non-referential ones from the use of zero pronouns which are frequent in pro-drop languages.

This paper focuses on identifying non-referential subject zero pronouns in Korean sentences. The proposed model learns structural information of clauses, and directly identifies non-referential uses using non-referential training instances. Our experimental results show that information of clauses are important to identify non-referential zero pronouns. Our method outperforms the baseline systems and the obtained results show that structural information of clauses plays a positive role in solving our task.

In the future, we plan to apply the proposed method to a practical Korean-English machine translation system. In addition, future work is needed to develop more advanced methods to determine the referentiality from imbalanced data.

Acknowledgement

This work was supported by the IT R&D program of MKE/IITA. [Development of a Cognitive Planning and Learning Model for Mobile Platforms].

References

1. Halliday, M.A.K., Hasan, R.: *Cohesion in English*. London Publishing Group (1976)
2. Grosz, B.J., Joshi, A.K., Weinstein, S.: Centering: A Framework for Modeling the Local Coherence of Discourse. *Computational Linguistics* 21(2), 203–225 (1995)
3. Haussler, D.: *Convolution Kernels on Discrete Structures*. UCS-CRL-99-10, UC Santa Cruz (1999)
4. Joachims, T.: Making large-Scale SVM Learning Practical. In: Scholkopf, B., Burges, C., Smola, A. (eds.) *Advances in Kernel Methods - Support Vector Learning*. MIT Press, Cambridge (1999)
5. Japkowicz, N.: The class imbalance problem: Significance and strategies. In: *The International Conference on Artificial Intelligence, Las Vegas* (2000)
6. Collins, M., Duffy, N.: Convolution Kernels for Natural Language. In: *Neural Information Processing Systems (NIPS)*, pp. 625–632 (2001)
7. Soon, W.M., Ng, H.T., Lim, D.C.Y.: A Machine Learning Approach to Coreference Resolution of Noun Phrases. *Computational Linguistics* 27(4), 521–544 (2001)
8. Evans, R.: Applying machine learning toward an automatic classification of it. *Literary and Linguistic Computing* 16(1), 45–57 (2002)
9. Kotsiantis, S.B., Pintelas, P.E.: Mixture of Expert Agents for Handling Imbalanced Data Sets. *Annals of Mathematics, Computing & Teleinformatics* 1(1), 46–55 (2003)
10. Ng, V.: Learning noun phrase anaphoricity to improve coreference resolution: Issues in representation and optimization. In: *42nd Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 152–159 (2004)
11. Iida, R., Inui, K., Matsumoto, Y.: Anaphora Resolution by Antecedent Identification Followed by Anaphoricity Determination. *ACM Transactions on Asian Language Information Processing* 4(4), 417–434 (2005)
12. Han, N.-R.: *Korean Zero Pronoun: Analysis and Resolution*. Doctoral dissertation, Department of Linguistics at the University of Pennsylvania (2006)
13. Moschitti, A.: Making Tree Kernels Practical for Natural Language Learning. In: *11th International Conference on European Association for Computational Linguistics*, pp. 113–120 (2006)
14. Roh, J.-E., Lee, J.-H.: Generation of Zero Pronouns Based on the Centering Theory and Pairwise Saliency of Entities. *IEICE Transactions on Information and Systems* E89-D(2), 837–846 (2006)
15. Iida, R., Inui, K., Matsumoto, Y.: Zero-Anaphora Resolution by Learning Rich Syntactic Pattern Features. *ACM Transactions on Asian Language Information Processing*, article 12, 6(4) (2007)
16. Zhao, S., Ng, H.T.: Identification and Resolution of Chinese Zero Pronouns: A Machine Learning Approach. In: *2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pp. 541–550 (2007)
17. Bergsma, S., Lin, D., Gorbel, R.: Distributional Identification of Non-Referential Pronouns. In: *ACL-HLT 2008, Columbus, Ohio*, pp. 10–18 (2008)
18. Kim, K.-S., Park, S.-B., Song, H.-J., Park, S.-Y., Lee, S.-J.: Identification of Subject Sharpness for Korean-English Machine Translation. In: *10th Pacific Rim International Conference on Artificial Intelligence*, pp. 211–222 (2008)
19. Yang, X., Su, J., Tan, C.L.: A Twin-Candidate Model for Learning-Based Anaphora Resolution. *Computational Linguistics* 34(3), 3270–3356 (2008)
20. Iida, R., Inui, K., Matsumoto, Y.: Capturing Saliency with a Trainable Cache Model for Zero-anaphora Resolution. In: *Joint Conference of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing*, pp. 647–655 (2009)
21. Wu, D., Liang, T.: Zero Anaphora Resolution by Case-based Reasoning and Pattern Conceptualization. *Expert Systems with Applications* 36(4), 7544–7551 (2009)

Identifying Idiomatic Expressions Using Phrase Alignments in Bilingual Parallel Corpus

Hyoung-Gyu Lee¹, Min-Jeong Kim¹, Gumwon Hong¹,
Sang-Bum Kim², Young-Sook Hwang², and Hae-Chang Rim¹

¹ Department of Computer and Radio Communications Engineering,
Korea University, Seoul, Korea

{hglee,mjkim,gwhong,rim}@nlp.korea.ac.kr

² Convergence Technology Center, SK Telecom

{sangbumkim,yshwang}@sktelecom.com

Abstract. Previous efforts to identify idiomatic expressions using a bilingual parallel corpus have focused on the method of using word alignments to catch the sense of individual words. In this paper, we propose a method of using phrase alignments rather than word alignments in a parallel corpus to recognize the sense of phrases as well as words. Our proposed scoring functions are based on the difference of translation tendency between a phrase and individual words. They can help us identify idiomatic expressions with a entropy variation and a translation difference between a phrase and individual words. Experimental results show that our proposed method is more effective than previous approaches for the identification of idiomatic expressions. In addition, we proved that linguistic constraints can be integrated into our method to improve the performance.

1 Introduction

An idiomatic expression is often defined as a sequence of words which has a different meaning from the composition of the meaning of its individual words, although it is difficult to find a universal definition that covers all kinds of typical idioms such as “kick the bucket” and “give up” [1]. In this paper, we regard idiomatic expressions as non-compositional expressions in the same manner as some previous works for the identification of idiomatic expressions [1,2,3].

Identifying idiomatic expressions is invaluable for natural language processing applications such as machine translation, information retrieval, and so on. Most rule-based machine translation systems generally translate idiomatic expressions prior to the word-for-word translation step in order to keep the adequacy in the first step. It is necessary to identify idiomatic expressions in a user query to improve the effect of the query expansion in information retrieval. Moreover, idiomatic expressions can be used as a significant unit when documents are indexed by terms.

Our task can be summarized as follows:

- Input: A sequence which contains two or more words.
- Output: A score that shows how much the input is idiomatic or non-compositional.

An expression with high score is more idiomatic than the one with low score. This definition is same as that of the task carried out in [3]. We are interested in scoring how close a word sequence is to an idiomatic expression.

Most previous efforts have used the statistical information from a corpus to identify idiomatic expressions. They are classified into two groups by the corpus type, which is either a monolingual or a bilingual corpus. Up to date, the approaches using monolingual corpora [1,4] are much more prevalent than efforts using bilingual corpora [3,5] due to the convenience of collecting the corpora.

However, statistical machine translation has been receiving increasing attentions over the last decade and has leded the production of bilingual parallel corpora available in various language pairs. For this reason, exploring the bilingual parallel corpora has become an interesting topic for researchers in order to extract useful knowledge such as paraphrases [6], bilingual or multi-lingual dictionaries [7,8].

The motivation of using bilingual corpus rather than monolingual corpus for idiomatic expression identification is as follows. By translating a multi-word expression, we can easily test whether it is an idiomatic expression or not. A word may be translated differently according to the idiomatic expressions it occurs in. If we cannot easily translate the combination word by word (with default translation¹), then that is strong evidence of an idiomatic expression. Nevertheless, there are not much work on identifying idiomatic expressions using bilingual parallel corpora.

The previous approaches [3,9] using bilingual corpora measured the translational entropy or the proportion of default translation of individual words in a given expression to rank given candidate expressions and to identify idiomatic expressions.

Although they have shown some promising results, there are two limitations using only word alignments. Firstly, the methods using word alignments can generate some errors in the process of calculating the translation entropy of a word or of extracting default translations of a word. A source word may be translated into more than one target word (one-to-many alignment) as well as exactly one word (one-to-one alignment). The word-based methods cause the problem that they measure the translational entropy imprecisely or extract the default word translation incorrectly, because an one-to-many alignment is regarded as multiple one-to-one word alignments rather than a single one-to-one phrase alignment. Secondly, the phrase-level translations are not considered in the previous methods, while they inspect only the word-level translation of expressions using word alignments. For identifying idiomatic expressions, we assume that it is important to analyze the difference of the translation tendency between a phrase and individual words in the phrase, which is not considered in previous approaches.

In this paper, we propose a method of using phrase alignments rather than word alignments in a parallel corpus to identify idiomatic expressions. In order to identify idiomatic expressions more precisely, we propose:

¹ The default translation of a word or a phrase means the most typical translation into the target language.

- examining a method of using phrase alignments instead of word alignments
- calculating the idiomatic expression score by new scoring functions based on the phrase alignments

The rest of this paper is structured as follows. In section 2, we propose our novel scoring functions for identifying idiomatic expressions and the method for phrase alignment. After that we evaluate the proposed method and analyze the results in section 3. We conclude the paper with some future works in section 4.

2 Phrase-Alignment Based Idiomatic Expression Identification

In this section, we present the intuitions of our method and proposed scoring functions to identify idiomatic expressions using a bilingual parallel corpus.

2.1 Finding Phrase Alignment

It is necessary to extract not only word-based properties but also phrase-based properties in a corpus for identifying idiomatic expressions because they are phrases - a sequence of two or more words. We propose a method of using phrase alignments for identifying these expressions in a bilingual parallel corpus. The phrase alignments provide useful statistics used to predict the translation tendency of a phrase.

The phrase alignment has been widely studied in the area of the statistical machine translation [10,11,12,13,14]. It aims to link a source phrase to a target phrase which is likely to be the translation of the source phrase in a given parallel sentence. Fig. 1 shows examples of word alignments and their phrase alignment. The black small boxes in the alignment table indicate word alignments in a English-Korean sentence pair, "john kicked the bucket" and

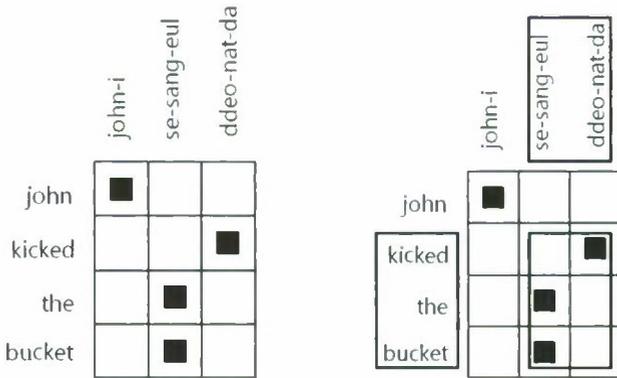


Fig. 1. Examples of Word Alignment and Phrase Alignment

“존이 세상을 떠났다 (john-i se-sang-eul ddeo-nat-da)”. And the large quadrangle including three word alignment boxes shows a phrase alignment in the right-side table. English phrase “kicked the bucket” is aligned with Korean phrase “세상을 떠났다 (se-sang-eul ddeo-nat-da)” in phrase level. These phrase-level approaches led to great improvement in statistical machine translation.

We adopt statistical phrase-based translation [10] to find phrase alignments in a bilingual parallel corpus. Although there are a variety of phrase alignment techniques, we use the method proposed by Och and Ney [11] among them. It is the most popular method which extracts all aligned phrase pairs from word alignment result. Phrase alignments by this method include one-to-one, one-to-many, many-to-one, and many-to-many word alignments.

2.2 Scoring Idiomatic Expression

We propose two novel scoring functions based on phrase alignments and the combination method of two functions. The functions commonly output the score which shows the degree of the closeness to idiomatic expressions, given a phrase as input.

DTE: Decrement of Translational Entropy. An idiomatic expression is a phrase which has a meaning that cannot be derived by decomposing it into its words. The translation of an idiomatic phrase tends to be limited to only a few target phrases, even though each word in the phrase may be translated as various words or phrases in the corpus. For example, Korean translations of English phrase “lie down” are significantly restricted to “눕다 (nup-da)” or “ㄷ 러눕다 (deu-reo-nup-da)”, while Korean translations of the word “lie” or “down” are various and evenly distributed.

Therefore, it is important to investigate a decrement of the translational entropy when individual words grouped together as a phrase. In other words, if the average translational entropy of individual words is high and the translational entropy of the phrase itself including them is low, it is more likely to be an idiomatic expression. The following equation reflects this idea.

$$Score_{DTE}(p) = \frac{1}{2} \left(\frac{\sum_{w \in W_p} H(T_w|w)}{|W_p|} + (1 - H(T_p|p)) \right) \quad (1)$$

where W_p is a set of words in the phrase p and T_p is a set of phrases aligned with p . $H(T_p|p)$ is the translational entropy [9] of p , which is calculated in the following equation:

$$H(T_p|p) = - \sum_{t \in T_p} P(t|p) \log P(t|p) \quad (2)$$

We select the base of the logarithm according to the size of T_p to normalize the entropy into the value between 0 and 1. This normalization allows the entropy to be comparable and $Score_{DTE}()$ to return the value between 0 and 1. $P(t|p)$ is

identical to the phrase translation probability estimated by the relative frequency of phrase pairs in statistical phrase-based translation [10].

$$P(t|p) = \frac{\text{count}(t, p)}{\sum_t \text{count}(t, p)} \quad (3)$$

For example, the scores of the literal phrase “tv drama” and the idiomatic phrase “new york” are calculated as follows. These examples show that our first function helps us distinguish idiomatic phrases from literal phrases.

$$\text{Score}_{DTE}(\text{“tv drama”}) = \frac{1}{2} \left(\frac{0.28 + 0.48}{2} + (1 - 0.73) \right) = 0.32 \quad (4)$$

$$\text{Score}_{DTE}(\text{“new york”}) = \frac{1}{2} \left(\frac{0.72 + 0.54}{2} + (1 - 0.19) \right) = 0.72 \quad (5)$$

DTW: Difference of Translated Words. In the second scoring function, we use the default phrase translations of words or phrases to recognize the meaning of them. A source phrase is most likely translated into the default phrase translation of it. For instance, an English phrase “give up” has the Korean default phrase translation “포기하다(po-gi-ha-da)” whose meaning is “to stop trying to do something”.

We assume that there exists larger translational difference between the phrase and individual words in an idiomatic phrase than in a literal phrase. The difference can be found by inspecting default word translation and default phrase translation. The following equation is the scoring function for quantifying the difference.

$$\text{Score}_{DTW}(p) = 1 - \frac{|W_{D_p} \cap \bigcup_{w \in W_p} W_{D_w}|}{|W_{D_p}|} \quad (6)$$

where D_p is a set of default phrase translations of the phrase p , i.e. N -best translations of p , and D_w is also N -best translations of the word w . The optimal N is empirically obtained by experiment. W_p is a set of words in p like the preceding. As the following equation shows, W_{D_p} and W_{D_w} mean sets of all words in D_p and D_w , respectively.

$$W_{D_p} = \bigcup_{d \in D_p} W_d \quad (7)$$

The denominator of equation 6 means the number of words in default translations of the phrase p . The numerator means the number of words which occur in both default translations of p and all default translations of the individual words. If the fraction is large, there are few differences between them. This indicates that p is close to a literal expression. We subtract the fraction from 1 to give high scores to idiomatic phrases.

The intuition of this scoring function is similar to that of the proportion of default alignment (PDA) proposed by previous work [3]. However, we directly extract default phrase translations using phrase alignments.

For example, the scores of the literal phrase “tv drama” and the idiomatic phrase “take charge of” are calculated as follows. These examples show that our second function also helps us distinguish idiomatic phrases from literal phrases. We assume that N is set to 2 in this example.

$$D_{tv} = \{tv, tel-le-bi-jeon\}$$

$$D_{drama} = \{deu-ra-ma, sa-geuk\} \quad (8)$$

$$D_{tv\ drama} = \{deu-ra-ma, tv\ deu-ra-ma\}$$

$$Score_{DTW}(tv\ drama) = 1 - \frac{3}{3} = 0.00 \quad (9)$$

$$D_{take} = \{chwi-ha-da, ha-da\}$$

$$D_{charge} = \{hyeom-cui, go\ it\}$$

$$D_{of} = \{cui, e\ dae-han\} \quad (10)$$

$$D_{take\ charge\ of} = \{reul\ mat, mat\}$$

$$Score_{DTW}(take\ charge\ of) = 1 - \frac{0}{3} = 1.00 \quad (11)$$

We derive the final scoring function in which two proposed functions are combined linearly as follows. The parameter λ is estimated empirically.

$$Score_{comb}(p) = \lambda Score_{DTE}(p) + (1 - \lambda) Score_{DTW}(p) \quad (12)$$

3 Experiments

3.1 Setup

We have experimented with an English-Korean parallel corpus to acquire English idiomatic expressions. The corpus, which includes about half a million sentence pairs, was collected from English-Korean bilingual news websites.² Table 1 shows statistics of the collections.

We automatically aligned source words with target words using the GIZA++ toolkit [15] in the corpus. We symmetrized the bidirectional results of word alignments using three types of heuristics; *intersection*, *union*, and *grow-diag-final*. All experimental results in this section are delivered from *grow-diag-final* because it reaches the best performance.

Next, we extracted phrase pairs from the word aligned corpus using the phrase extraction algorithm proposed by Och and Ney [11] and estimated the translation probability of every unique phrase pair by calculating the relative frequency of phrase pairs. The translation probabilities are used to calculate the phrase translational entropy and to find default phrase translations of phrases. We extracted N -best phrase alignments with high translation probability for each phrase in the corpus in advance to use as default phrase translations. N is set as 2 experimentally.

It is necessary to construct a set of test phrases to evaluate the proposed method. Also, each test phrase should have the gold annotation that indicates

² It is a part of the resources from on-going project sponsored by SK-telecom, Korea.

Table 1. Corpus Statistics

	English	Korean
Training Sentences	493,000	
Words/Morphemes	10,857,668	12,868,977

whether it is an idiomatic expression or not. The candidate phrases for the evaluation may be collected using various heuristics or linguistic constraints. For example, VP-PP tuples were used as test phrases in previous work [3]. Our evaluation focus on the scoring function for identifying idiomatic expressions in a set of candidate phrases rather than the extraction of candidate phrases. For this reason, we simply extracted candidate phrases using phrase extraction algorithm and several constraints. Our every candidate phrase occurs three or more times in the first 200,000 sentences and involves two or more content words. We sampled 300 phrases from all candidate phrases and then two annotators manually annotated all idiomatic expressions in the phrase set. Among them 55 phrases were annotated as idiomatic expressions by both annotators. The inter-annotator agreement for these annotations was measured at 0.863 agreement rate and 0.638 kappa value.

We used *average precision* to evaluate the ranked result. The evaluation measure, which is frequently used in information retrieval field, emphasizes ranking relevant items higher:

$$AveP = \frac{\sum_{r=1}^N (P(r)) \times rel(r)}{\text{number of relevant items}} \quad (13)$$

where r is the rank, N is the number of retrieved items, $rel()$ is an indicator function on the relevance of a given rank, and $P(r)$ is precision computed at the point of the rank. In our case, candidate phrases and idiomatic expressions correspond to items and relevant items, respectively.

3.2 Experimental Results

Baseline. We implemented *translational entropy* (TE) and *proportion of default alignment* (PDA) proposed by Melamed [9] and Moiron [3] respectively as baselines compared with our proposed method.

Table 2 shows the performances of the identification of English idiomatic expressions using TE and PDA. Fig. 2 shows the combination performances

Table 2. Performances with TE and PDA

Alignment Type	Scoring Function	AveP	P@20	P@30	P@55
Word Alignment (baseline)	TE ($\lambda = 1$)	0.312	0.450	0.333	0.291
	PDA ($\lambda = 0$)	0.244	0.250	0.267	0.291

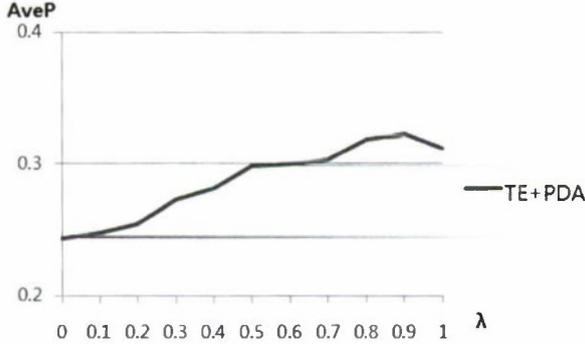


Fig. 2. Average Precision of TE + PDA according to lambda

Table 3. Effect of DTE and DTW

Alignment Type	Scoring Function	AveP	P@20	P@30	P@55
Word Alignment (baseline)	TE+PDA ($\lambda = 0.9$)	0.323	0.350	0.333	0.273
Phrase Alignment (proposed method)	DTE ($\lambda = 1$)	0.341	0.400	0.333	0.364
	DTW ($\lambda = 0$)	0.440	0.650	0.600	0.491
	DTE+DTW ($\lambda = 0.5$)	0.508	0.650	0.633	0.473

of two approaches according to the weight lambda. The best performance was obtained when the weight was set to 0.9. We use this figures as a baseline for our study.

Effect of DTE and DTW. Table 3 shows the performance for English idiomatic expressions identification in an English-Korean parallel corpus. The first row is the baseline and the followed three rows are the results by our proposed scoring functions. Both two proposed functions DTE and DTW achieved better performances than the baseline. This result shows that examining phrase alignments produce positive effects and proposed functions improve the performance of idiomatic expressions identification.

DTE is a phrase-level extension of TE and DTW is a phrase-level extension of PDA. In terms of these extensions, we found that both DTE and DTW are more effective in idiomatic expressions identification than TE and PDA, respectively and the latter brought about larger effects than the former. This shows that the use of default phrase translations, which was not considered in the baseline approaches, is very useful.

However, the reason why DTE produces disappointing performances is found in the phrase translational entropy calculation step. There are many target phrases with same topic and different expressions in a set of translated phrases of an source phrase. Such different target phrases with same meaning propagate

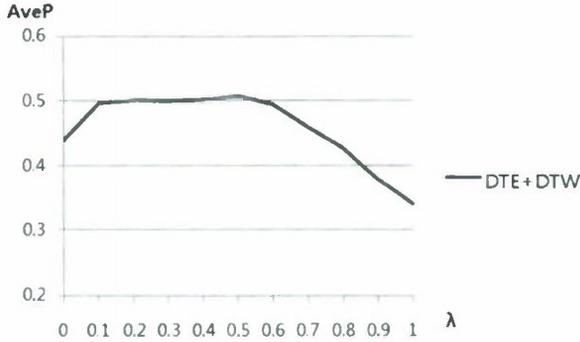


Fig. 3. Average Precision of DTE + DTW according to lambda

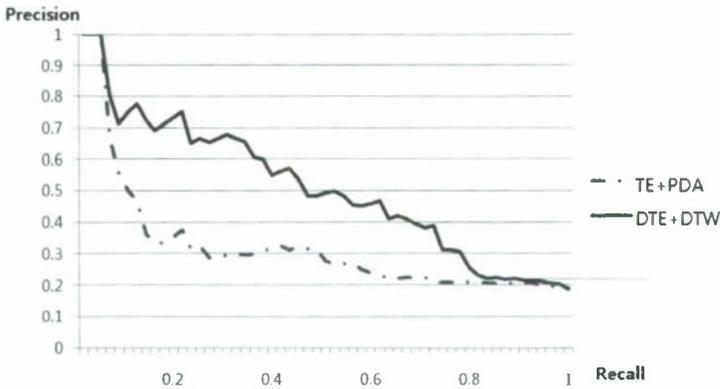


Fig. 4. Recall-Precision Graph of Previous and Proposed Scoring Functions

many errors into the entropy of each phrase. We expect to minimize these errors by clustering target phrases aligned with the source phrase in the future.

We also observed that DTW is complementary to DTE by combining the two functions. The last row of Table 3 shows the effect of this combination. This is because DTE identify idiomatic proper nouns such as “new york” or “korea university” more accurately than DTW, while DTW recognize idiomatic verb phrases or prepositional phrases better than DTE. Fig. 3 shows the average precision of our proposed method according to the parameter lambda. The best performance was obtained at 0.5.

Fig. 4 is the recall-precision graph of the baseline and the proposed method. The x-axis and the y-axis indicate the recall and the precision, respectively. The method using phrase alignments has higher precision at overall recall levels than the method using word alignments. Besides, we found that there is a large gap of the precision in 0.2-0.4 recall levels between two approaches, while they are similarly effective in 0-0.1 recall levels.

Table 4. Effect of Linguistic Constraint

Alignment Type	Scoring Function	AveP	P@20	P@30	P@55
Phrase Alignment	DTE+DTW	0.508	0.650	0.633	0.473
(proposed method)	DTE+DTW+Constraint	0.519	0.700	0.633	0.509

Further Improvement with Linguistic Constraint. So far, we have presented the method independent of any language pairs. Now we prove that some linguistic constraints can be integrated into the method to improve the performance of idiomatic expression identification. In this experiments, we simply added two rules to the scoring process as follows.

- Rule 1: Exclude English articles such as “a” or “the” from averaging translational entropy values of individual words in a phrase in DTE.
- Rule 2: Exclude Korean functional words such as postpositions and endings e.g. “을 (eul)”, “으로 (eu-ro)”, or “에서 (e-seo)” from W_p in DTW.

We expect that Rule 1 will be effective for our task because English articles are usually not translated to any Korean words in English-Korean translation. Rule 2 is under the assumption that the non-compositionality of words does not rely on the difference of functional words in translated phrases of the source words. The figures in Table 4 imply that these techniques are valuable for our approach.

4 Conclusion and Future Work

This paper proposed a method for identifying idiomatic expressions using phrase alignments instead of word alignments in a bilingual parallel corpus. In this work, we focused on overcoming the limitations of previous approaches and quantifying the difference of the translation tendency between a phrase and individual words in the phrase. We proposed two scoring functions in which such differences reflected. The experimental results showed that our proposed scoring functions was effective in idiomatic expressions identification. Moreover, we presented that linguistic constraints can be integrated into our method to improve the performance.

For the future work, we first intend to explore the method using not only English-Korean but also English-French or English-Chinese parallel corpora together, in order to identify English idiomatic expressions. Secondly, we plan to identify Korean idiomatic expressions by changing only the translation direction from English-Korean to Korean-English. Also, we intend to improve the quality or the efficiency of machine translation systems with the idiomatic expressions identified by our approach.

Acknowledgement

We would like to thank members of the KUNLP MT project, Joo-Young Lee, Yeon-Soo Lee, Seung-Wook Lee, Jae-Hee Lee, and Ying-Xiu Quan, for their continuous advice in our work.

References

1. Fazly, A., Cook, P., Stevenson, S.: Unsupervised type and token identification of idiomatic expressions. *Computational Linguistics* 35(1), 61–103 (2009)
2. Li, L., Sporleder, C.: Classifier combination for contextual idiom detection without labelled data. In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 315–323 (2009)
3. Moiron, B.V., Tiedemann, J.: Identifying idiomatic expressions using automatic word alignment. In: *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics 2006 Workshop on Multiword Expressions*, pp. 33–40 (April 2006)
4. Lin, D.: Automatic identification of non-compositional phrases. In: *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pp. 317–324. Association for Computational Linguistics (June 1999)
5. Melamed, I.D.: Automatic discovery of non-compositional compounds in parallel data. In: *Proceedings of 2nd Conference on Empirical Methods in Natural Language Processing (EMNLP 1997)*, Providence, RI (1997)
6. Bannard, C., Callison-Burch, C.: Paraphrasing with bilingual parallel corpora. In: *ACL 2005: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, Morristown, NJ, USA, pp. 597–604. Association for Computational Linguistics (2005)
7. Wu, D., Xia, X.: Learning an english-chinese lexicon from a parallel corpus. In: *Proceedings of the First Conference of the Association for Machine Translation in the Americas (1994)*
8. Fung, P.: A statistical view on bilingual lexicon extraction: From parallel corpora to non-parallel corpora. In: *Parallel Text Processing*, pp. 1–17. Springer, Heidelberg (1998)
9. Melamed, I.D.: Measuring semantic entropy. In: *Proceedings of the SIGLEX Workshop on Tagging Text with Lexical Semantics: Why, What, and How*, Washington, pp. 41–46 (1997)
10. Koehn, P., Och, F.J., Marcu, D.: Statistical phrase-based translation. In: *NAACL 2003: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pp. 48–54. Association for Computational Linguistics (2003)
11. Och, F.J., Tillmann, C., Ney, H.: Improved alignment models for statistical machine translation. In: *Proceedings of the Joint Conference of Empirical Methods in Natural Language Processing and Very Large Corpora*, pp. 20–28 (1999)
12. Marcu, D., Wong, W.: A phrase-based, joint probability model for statistical machine translation. In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 133–139 (2002)
13. Zhang, Y., Vogel, S.: An efficient phrase-to-phrase alignment model for arbitrarily long phrase and large corpora. In: *Proceedings of the Tenth Conference of the European Association for Machine Translation, EAMT 2005 (2005)*
14. DeNero, J., Klein, D.: The complexity of phrase alignment problems. In: *HLT 2008: Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies*, Morristown, NJ, USA, pp. 25–28. Association for Computational Linguistics (2008)
15. Och, F.J., Ney, H.: A systematic comparison of various statistical alignment models. *Computational Linguistics* 29(1), 19–51 (2003)

Generating an Efficient Sensor Network Program by Partial Deduction

Li Li¹ and Kerry Taylor²

¹ Southwest University, Chongqing, P.R. China
lily@swu.edu.cn

² ICT Centre, CSIRO, Australia
kerry.taylor@csiro.au

Abstract. Partial deduction is an optimisation technique developed by the logic programming community. We propose the use of Partial deduction in the domain of wireless sensor network programming where programs are written for small computational platforms and energy is typically scarce. We show how, together with a declarative programming language which has been shown to be suitable for several demanding sensor network applications, it can address key issues such as rewriting a query using views and reducing redundancy of rewritings as long as some computation and abstraction can be performed at compile-time, which obviously leads to the improvement of energy efficiency at run-time. We argue that energy efficiency can be achieved with: (1) minimised sensor network programming workload by forcing the folding of goals into the view partially; (2) reduced redundant computation with fewer computation steps at network nodes by forcing the unfolding of simple goals; (3) reduced inter-node message transmission by more specific addressing of messages to nodes; and (4) reduced memory requirements by specialising network-wide programs to smaller programs for specific nodes. A partial deduction system is developed and an extended example is provided to demonstrate the potential performance improvement of the technique.

1 Introduction

Wireless sensor networks (WSNs) promise to revolutionise sensing in a wide range of application domains. They can be used to offer the potential to advance scientific pursuits in areas such as manufacturing, agriculture, and transport [1]. However, wide acceptance and deployment has not yet occurred because of lack of robust of platforms and lack of fully functional support for data manipulation. From a technical point of view, one may think of a sensor network as a database that is able to conduct query processing, which includes a large range of heterogeneous data distributed arbitrarily. Other than that in a traditional DBMS, query processing works differently to that in a sensor network because changes in sensor networks may happen unpredictably to the data collection regime as sensors come and go in addition to the imperfect link quality. Furthermore, the sensed events and sensing intervals may vary dramatically on different occasions,

and the volume of the sensing can be very variable depending on sampling rate variations. Traditional database optimisation techniques of specifying join methods and indices are still useful but the unique characteristics of sensor networks should be considered as a query in sensor networks may be based either on live data or archived data or a mix of both of them. As for archived data, we are interested in using a set of views¹ V expressed in terms of archived data sources to associate with previous query results. Roughly, the following steps are needed to process the query if the end-use would like to “find sensors (i.e. locations, sensor IDs) where the temperature measurements are within a specific range X and their residual power at least Y units, and send the new temperature to its available neighbours”. Including:

1. decide if the query can be fully answered by using views V
2. if not, (using views as many as possible) develop a sensor network program (in a logic programming language) with respect to the query
3. (applying a dedicated optimisation technique) generate an efficient sensor network program from step 2) to cope with severe resource and bandwidth constraints on the sensor nodes

Usually, a sensor network query will ask for live sensor readings. Therefore to provide a solution to the last two steps is necessary, and this will be the main focus of this paper. Specifically, for a query expressed in a logic programming language, we are looking into rewriting this query using views first and then specialising this network-wide program to a smaller program for the specific nodes rather than all nodes. We propose using partial deduction to achieve the goal. In order to meet our requirement, the design of fold/unfold control will be considered. Particularly, we are interested in problems that either part of their definitions (e.g. the code to solve them) are available or bindings of variables can be computed at compile-time as this sort of problems will benefit considerably from partial deduction. In saying so, this paper makes the following contributions:

- using views to rewrite a query for sensor network query processing is discussed;
- fold/unfold control to generate a compact new program is investigated;
- a generic partial deduction system to generate a smaller program is developed;
- the cost analysis to show the significant difference by applying partial deduction is given.

In order to ensure that partial deduction to make good use of the logic structure of a problem and other data sources, essentially we need an expressive language to describe a broad range of problems but restrictive enough to allow efficient algorithms to operate over it. In fact, as pointed out in [2,3,4,5,6], it is natural to choose a declarative language to describe problems (e.g. queries) as it is offers an easy-to-understand programming interface. Moreover, it opens up the

¹ Views are simply results from previous queries. Prolog-style notation is used throughout the paper for views and queries.

possibility for optimisation algorithms to handle for the efficient access strategies transparent to the user. As a result, we will use a logic programming language (e.g. \mathcal{L}) throughout the paper. From a programming perspective, we will not differentiate wireless sensor network (WSN) programming from sensor network (SN) programming in this paper.

The rest of the paper is organised as follows. Section 2 introduces the important definitions and background. Section 3 discusses partial deduction to generate an efficient sensor network program. Section 4 details the proposed optimisation technique with an extended example followed by the cost analysis. Section 5 briefly reviews the related work. Section 6 presents the conclusion and future work.

2 Preliminary

In this paper, query processing aims to generate an efficient sensor network program with respect to a specific query. Informally, if we have:

- a query Q expressed in the language \mathcal{L}
- a set of views V expressed in terms of archived data source S also in \mathcal{L}
- a generic sensor program in the same language

and we want to generate a new program (i.e. **NewPgm** in Fig. 1) with respect to the original one (i.e. **Pgm**), the development of the partial deduction system is critical to the success of query processing.

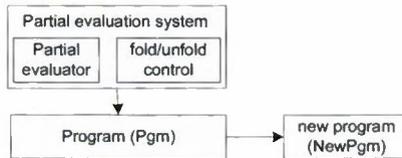


Fig. 1. Generate an efficient program

Following are some definitions to better understand partial deduction. Refer to the logic programming literature [7,8] for more detailed definitions.

Definition (clause). A clause is a disjunction of literals. In first-order logic, a clause is the universal quantification of all free variables of a quantifier-free disjunction of literals. Formally, a first-order literal is formula of the kind of $P(t_1, \dots, t_n)$ or $\neg P(t_1, \dots, t_n)$, where P is a predicate of arity n and each t_i is an arbitrary term. A clause is usually written as the implication of a head from a body. In this paper, we consider clauses with at most one positive literal.

Definition (conjunctive query). A conjunctive query has the form $H(\vec{X}) : -B_1(\vec{X}_1), \dots, B_m(\vec{X}_m)$, where $H(\vec{X})$ is a head, $B_1(\vec{X}_i)$ is a sub-goal in the body, and the tuple \vec{X}_i contains either variables or constants. All queries are required to be safe, i.e., that $X \subseteq \vec{X}_1 \cup \dots \cup \vec{X}_m$.

Definition (views). A set of view definitions (e.g. clauses) have the same form (i.e. represented by the head and body) but expressed in terms of a set of database relations. For example, we have the view v_1 in a form of $v_1(\text{Src}) : -\text{residualPower}(@\text{Src}, Y), Y > 1000$. It means that v_1 stores all sensors (Ids) with the residual power greater than 1000 units.

Definition (program). A program is a finite set of definite clauses.

Definition (unifier). A unifier of two terms is a substitution making the terms identical. If two terms have a unifier, they are said to unify. Further explanation is given subsequently.

Definition(unification). Unification is performed between the predicates and the atoms or terms in a program. If a unification succeeds, that is, the predicate names, arity (i.e. the number of arguments), and arguments are the same, the variables (the binding of the variables) will be instantiated.

Definition (unfolding). Substituting a goal in the body of a clause by the corresponding body. For example, unfolding a sub-goal B_i in a clause $H \leftarrow B_1, \dots, B_n$ with respect to a clause $B \leftarrow C_1, \dots, C_m$ where B and B_i unify with Θ , produces a clause: $(A \leftarrow B_1, \dots, B_{i-1}, C_1, \dots, C_m, B_{i+1}, \dots, B_n)\Theta$. Unfolding propagates bindings. In this paper, unfolding is also called unification-based propagation.

Definition (folding). The inverse of unfolding, whereby an instance of a predicate is substituted by the corresponding call. More discussion is available in Section 4.

Definition (partial deduction). A system for controlled folding/unfolding is known as partial deduction. It is often used for specialising a program with respect to the incomplete input.

We are developing solutions to handle query processing in sensor networks. Early work which discusses query rewriting algorithms [9] and semantic sensor network service framework design [10] have been reported and they are integral parts of query processing. However, we will focus on using partial deduction to optimise a sensor network program in this paper.

3 Partial Deduction and Its Impact

Partial deduction is especially useful for removing levels of interpretation [11] to generate a specialised program that generally does far more efficiently than the generic program. This specialised program is consciously tailored to a particular task. The theoretical underpinnings of this approach were discussed in [12,8]. The main idea of partial deduction is to recursively perform fold/unfold until no more progress can be achieved [11]. A few things must be included to build such a kind of system. Generally we consider:

1. The residual program, which is equivalent to the original ones, should be kept.
2. Clauses must be handled as well as goals. Folding the head and unfolding the body are highly desired.

3. A few declarations must be made explicitly, for example, which goal(s)/sub-goal(s) should be folded, unfolded or left alone.

(a) if folding is required, to what they should be folded (to make good of using views as many as possible). Usually, it requires unfolding first (bindings propagation) to allow more specific folding.

(b) unfolding rules are required to control unfolding. It is the inverse of folding.

In addition to the general meta-interpreter discussed in [11] and the unfold criteria in [13], it is preferable to consider the unique characteristics of a sensor network while defining folding/unfolding rules, for example, we need to consider the sensor network (programming language) built-in predicates.

(c) the empty goal and `true` will be handled after both (a) and (b) have been performed recursively in the obvious way.

After such a system has been developed, it is expected that the `NewPgm` is smaller than the original (also generic) sensor program (i.e. `Pgm`). In addition, it has potential to reduce the size of the message and (total) data transmissions as well. They are illustrated by the following examples.

Example 1. Suppose there are two sensor nodes in the network, @1 and @2, respectively. The notation “@” in @*Id* means the host of the tuple at *Id*. For example, one single rule $q(@2, c, 1)$ is hosted at the node with $Id = 2$. There is another rule hosted at node 1: $p(@1, X, Y) : -q(@2, X, Y), Y = \backslash = 1$. Fig. 2 shows the differences between two cases (1) and (2).

(1) without partial deduction (Fig. 2(a)): at least two variables X and Y are required to send from node 2 to node 1 to be evaluated at node 1.

(2) after partial deduction (Fig. 2(b)): as the fact at node 2 specifies the variable Y to be instantiate to 1, the second rule will not be fired. As a result, no variable is required to be transmitted as did in Fig. 2(a).



Fig. 2. Impact of partial deduction(1)

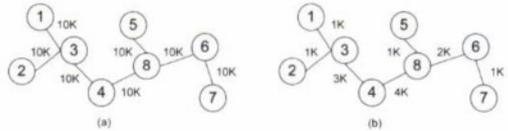


Fig. 3. Impact of partial deduction(2)

Example 2. Suppose there are eight nodes distributed as shown in Fig. 3(a), and the original program is about 10K bytes in size. In Fig. 3(a), the total data transmission is about 70K bytes in size. Suppose after applying partial deduction, the program was specialised to 1Kb for each of nodes. Then the total data transmission is about 13K as shown in Fig. 3(b), which is obviously less than 70K. Apart from reducing data transmission, partial deduction also plays an important role in code decomposition as one particular code was generated with respect to each of nodes by having taken account of some computation and abstraction at that node.

4 Query Processing and Cost Analysis

In this section, we will first look into an extended sensor network example to gain insight on partial deduction. We then present the cost analysis which clearly indicates that partial deduction has potential to improve query processing in sensor networks. Before we describe partial deduction in more detail, we give a brief description of the generic sensor network program used throughout the paper.

4.1 Running Example

the original program (Pgm):

```

-----
result(@Next,Src,Val,NewCost) :-                %... (1)
  message(@Src,Next,Dest,Val),
  timer(@Src,3,TimePeriod),
  candidate(@Src,Dest,Next,NewCost,NewHops).
timer(@Src,TimerID,TimePeriod) :-             %... (2)
  timerx(@Src,TimerID,TimePeriod).
message(@Src,Next,Dest,Val) :-                 %... (3)
  sensorId(@Src,Id),
  sensorMeasure(@Src,Id,Val1,Val2),
  range(ReqVal1,ReqVal2),
  Val1=<ReqVal1,ReqVal2=<Val2,
  reading(@Src,Id,H:M:S,Val),
  residualPower(@Src,Z),Z>1000.
sensorId(@Src,Id):-                             %... (4)
  sensor(Src),
  Id = Src.
reading(@Src,Id,H:M:S,Tval) :-                 %... (5)
  sampling(@Src,TimePeriod,TimePeriod,H:M:S,Tval).
candidate(@Src,Dest,Next,NewCost,NewHops) :-  %... (6)
  beacon(@Src,NewNext,Dest,OldNext,DldCost,DldHops),
  nextHop(@Src,Dest,_,_,_),
  linkLq1(@Src,NewNext,LinkCost),
  DldNext =\=Src,
  NewCost=DldCost+LinkCost,
  NewHops=DldHops+1,
  dest(@Src,Dest),
  Src=\=Dest.
-----

```

Following is a brief explanation of predicates in the elauses (1)~(6).

- The predicate *result/4* can be interpreted as: if there exist a message (i.e. *message/4*) and a candidate (i.e. *candidate/5*) whose next hop is the *Next*, and if the timer fires, then sends the message to the *Next*.
- the predicate *message/4* is further defined to symbolise the message by describing where the tuple will be sent to (i.e. *Next*), the origin of the message (i.e. *Src*), and the destination (i.e. *Dest*). The variable *Val* is the content of the message. It is the current temperature reading. However, the sensing (i.e. *reading/4*) will not take place until it is certain that this sensor has the capability to obtain it (that is, within the required range) and the node residual power is greater than 1000 mits.
- the predicate *sensorId/2* is defined to link *Src* and *Id* together, with *Id* specifying which sensor is currently concerned.

- the predicate *residualPower/2* measure the residual power at the node.
- the predicate *reading/4* is defined to present the sensing data with a given sampling rate within a sampling period.
- the predicate *sampling/5* is defined as a built-in predicate. To be brevity, the sampling rate and sampling period are set to same value in this program.
- the predicate *candidate/5* can be understood as a candidate to receive a new beacon message with *NewCost* and *NewHops* to be updated accordingly.
- the predicate *nextHop/5* is defined as another built-in predicate to indicate the next hop that the message should head for.
- the predicate *linkLqi/3* is also defined as a built-in predicate to represent the last received packet from the source (i.e. @Src).

We will use the view v_1 introduced in Section 2 to rewrite part of the clause (3). We are aware that different algorithms [9,14] for query rewriting exist. In this paper, we use the equivalent rewriting algorithm for demonstration.

Now let us revisit the sample query introduced in Section 1. The query is about “find sensors where the temperature measurements are within a specific range X and their residual power at least Y units, and send the new temperature to its available neighbours”. The predicate *find/4* is defined to represent the goal at an abstract level. The developed partial deduction system consists of two parts: the partial reducer and fold/unfold control. The following code snippet gives a brief idea of how a partial reducer looks like.

```
partial reducer:
-----
do_fold(H1,H2) :-
    fold(H1,H2), !.
do_unfold((H:-B),(H:-NB)) :-
    conjunct_to_list(B,BL),
    unfold(BL,NBL),
    list_to_conjunct(NBL,NB).
.....
-----
```

Note that as unfolding first allows more specific folding [11], we have to consider in which order the fold/unfold rules to be fired. In the example, all sub-goal(s) in the original program *Pgm* should be unfolded, while *result/4* should be folded into *find/4*. Following is a fragment of fold/unfold control in our example.

```
fold/unfold control:
-----
%variable 'Clauses' are clauses from Pgm
program(Pgm,Clauses).
%unfold all sub-goals from Pgm
unfold(message(@Src,Next,Dest,Val),Nm).
.....
%fold result/4 to find/4
fold(result(@Next,Node1,Val,NewCost),
    find(@Src,Node1,Val,NewCost)).
-----
```

Putting the preceding partial reducer and fold/unfold control together, the *Pgm* will be specialised into the following new program *NewPgm* (Note that *Node1* is 12, *Node2* can be either 11 or 15), given the view v_1 . We illustrate $Node1 = 12, Node2 = 11$ only.

NewPgm (the variables have been renamed by the system):

```
-----
find(@12,12,11,_G1413) :-
  sampling(@12,_G1422,_G1422,_G1429:_G1432:_G1433,_G1410),
  nextHop(@12,0,_G1440,_G1441,_G1442),
  linkLqi(@12,_G1447,_G1416).
-----
```

This new program entirely replaces the original one given earlier. It is specialised from clauses (1)~(6). Clearly, the specialised code is more compact. This is because among available nodes, only nodes {12}, {11, 15} meet the requirements, given the view v_1 . Other advantages will be discussed in the subsequent section.

4.2 Cost Analysis

For the cost analysis, we first analyse the cost matrix in our example. Since our focus is on query processing, only query related cost will be taken into account in this paper. We will consider data acquisition and transmission in the future work. Thus, the estimated cost is defined as the combination of:

- (c_{node}) rule evaluation associated cost at each of nodes
- (c_{trans}) the number of variables transmitting between two nodes

Following notations are used to simplify the analysis. They are:

- r - the number of rules/clauses in a program;
- l_i - the number of predicates in rule r_i ;
- d_j - the number of variables transmitting between two nodes when one predicate is concerned (see Fig. 2(a) for example).

We assume \sqrt{n} by \sqrt{n} square grid topology for the analysis. The basic idea is that we are able to count the number of transmissions with $\sqrt{n} - 1$ hops at most (e.g. diagonal routing), which is the longest path from one end to another. We also assume that there are m sensors in the network. In our cost model, the total number of predicates is defined as: $p = \sum_{i=1}^r l_i$(f1)

with these notations, c_{cost} is given below

$$c_{node} = m \times \sum_{i=1}^r l_i \tag{f2}$$

and the cost of variable transmission is defined as:

$$c_{trans} = m \times \sum_{i=1}^p d_j \times (\sqrt{n} - 1) \tag{f3}$$

Substituting p in formula (f3) by formula (f1), the total cost would be:

$$c_{total} = c_{node} + c_{trans} = m \times (\sum_{i=1}^r l_i + (\sqrt{n} - 1) \times \sum_{j=1}^{\sum_{i=1}^r l_i} d_j) \tag{f4}$$

The influence of partial deduction on c_{total} is obvious when the number of rules (i.e. r in the formula (f4)) to be fired was reduced. Moreover, in most cases, instead of all nodes to be involved in query processing, only relevant nodes specified by the specialised program will be active. As such, the number of the involved sensors, m , decreases tremendously. Consequently, c_{total} in formula (f4) will be reduced accordingly.

The detailed cost analysis based on the preceding example is given in Table 1. For simplicity, all predicates are treated equally in the table. The minimum number of variables in a predicate, say, 2, is used for the analysis. Based on the analysis shown in Table 1, it is clear that a significant difference in cost between the *Pgm* and *NewPgm* exists. We employ a metric, called *Diff*, to quantify the cost savings in query processing as a result of partial deduction. The *Diff* is computed as the difference between the sum of the cost of *Pgm* and *NewPgm*. An estimation is given as follows.

$$\begin{aligned} Diff_total &= c_total(Pgm) - c_total(NewPgm) \\ &= 27 \times m + 54 \times m \times (\sqrt{n} - 1) - 22 \times (\sqrt{n} - 1) - 8 \\ &= 27 \times m + 43 \times m \times (\sqrt{n} - 1) + 11 \times m \times (\sqrt{n} - 1) - 22 \times (\sqrt{n} - 1) - 8 \quad \dots(f5) \end{aligned}$$

Note that $m \gg 2$ in this example, thus, the second and third terms of (f5) can be removed if we simply let $m = 2$ for the term " $11 \times m \times (\sqrt{n} - 1)$ ", then we have

$$\begin{aligned} Diff_total &> 27 \times m + 43 \times m \times (\sqrt{n} - 1) - 8 \\ &> 26 \times m + 43 \times m \times (\sqrt{n} - 1) + 27 \times m - 8 \\ &> 26 \times m + 26 \times m \times (\sqrt{n} - 1) \\ &> 26 \times m \times \sqrt{n} \\ &> m \times \sqrt{n} \end{aligned}$$

Thus, the "order" of the calculation in the Big O notation is $O(m\sqrt{n})$ ($m \gg 2$, $\sqrt{n} > 1$).

Table 1. Cost analysis

Cost criteria	Pgm	NewPgm
r	6	1
p	27	4
No. of sensors	$m \gg 2$	2
c_cost	$27 \times m$	8
No. of variables	$> 27 \times 2 = 54$	11
c_trans	$54 \times m \times (\sqrt{n} - 1)$	$22 \times (\sqrt{n} - 1)$
c_total	$27 \times m + 54 \times m \times (\sqrt{n} - 1)$	$22 \times (\sqrt{n} - 1) + 8$

With partial deduction, generally, we have achieved:

- the new program is smaller and more compact than the original one
- the storage on nodes has been reduced as only fewer nodes need to consider it
- inter-node message transmission (i.e. variables transmission) has been reduced by more specific addressing of messages to nodes.

All these would make the improvement of the execution performance possible due to the computation and space complexity having been reduced.

5 Related Work

Following are brief overviews of the related work in sensor network query. Three categories are identified below. Let us first start from database query.

5.1 Database Query

TinyDB [15,16] (<http://telegraph.cs.berkeley.edu/tinydb/>), a seminal first-generation SN database, was developed by UC Berkeley. TinyDB's structure allows queries to be parsed and optimised at the base station. The optimisation phrase is focused on choosing the correct ordering of sampling, selections, and joins with the help of metadata [16]. However, little or no work of partial deduction has been reported and it is most likely that the non-specialised binary format of the queries are sent into the sensor network, where they are instantiated. This contrasts dramatically with our approach where the instantiation is performed at the sink and abstraction is performed with unification-based propagation throughout the program at compile-time before the specialised program to be distributed to the sensor network.

Cougar [4,17] (<http://www.cs.cornell.edu/database/cougar/index.php>) discusses queries over sensor networks by allowing users to task the network by adding a query layer above the networking layer in the protocol stack [5]. In-network aggregation is the focus of the paper. Again, known knowledge has not yet be discussed sufficiently. To our knowledge, none of TinyDB and Cougar has fully taken advantage of the knowledge known in priori explicitly in query processing.

Campton's paper [9] investigated a maximal rewriting using views in the presence of functional dependencies and value constraints. It will be studied further in our work.

5.2 Query Programming Language and Platform

Efforts from UC Berkeley present the design and implementation of a declarative sensor network platform (DSN) [2] which include a declarative language (i.e. Snlog), compiler and runtime which is supported by TinyOS (<http://www.tinyos.net/>). At the core of the platform lies the Snlog compiler that transforms the Snlog specification into nesC language which is native to TinyOS. The generated codes, together with relevant compiler libraries, are further compiled by the nesC compiler into binary image to injected into the nodes in the network. The focus of the DSN is on providing a single high-level programming environment. Authors in [2,18] have addressed traditional sensor network protocols and demonstrated that DSN is a natural fit for sensor networks. However, there is no discussion about implementing an efficient query processing by taking advantage of the known resources. We attempted to address this issue to reduce computation and bandwidth usage and eventually minimising data transmission for the resource constrained WSNs.

The SNEEql (Sensor NEtwork Engine query language) query optimiser [19] (<http://intranet.es.man.ac.uk/img/dias-mc/sneeqql-overview.php>) is a recent attempt which combines an expressive query language with a layered architecture to generate an executable nesC code. However, the proposed approach not seem to consider how to generate an efficient sensor network program.

5.3 SensorWeb

Other relevant work comes from sensornet (<http://www.sensornet.gov/>) and sensorWeb [20], where the knowledge known a priori has been used, either as an ontology or a repository, to improve query processing at the service level. Reported work has proposed a service-oriented framework to handle both data streams from WSNs and information retrieval requirements. These projects have different views about WSNs and none of them attempted to consider the efficient sensor network program generation in WSNs.

As discussed, we are interested in rewriting query using views and then reducing redundancy to generate an efficient sensor network program. We have demonstrated that partial deduction has potential to improve the application performance.

6 Conclusion

We argue that efficiency of a sensor network program can be improved with partial deduction by using views. We highlighted the significance of partial deduction in query processing. We have demonstrated that redundancy can be reduced considerably as long as some computation and abstraction can be performed at compile-time.

We are aware of the inherent limitations of partial deduction, but for a class of problems, we argue that they are much more gainful from partial deduction if either part of their definitions (the code to solve them) are available or bindings of variables can be computed at compile-time. We have demonstrated that for these kinds of problems, the advantages of partial deduction greatly outperforms its disadvantages. The analysis also suggests that it is promising to apply the proposed optimisation technique to reduce redundancy to better address tight energy and bandwidth issues in sensor network applications.

It is generally expected that an automatic sensor network program generator is developed to lessen the heavy burden of rewriting an arbitrary query. In order to achieve this goal, we plan to study sensor network query rewriting in depth in the future.

Acknowledgments

The authors would like to thank David Ratcliffe and Doug Palmer from CSIRO ICT centre Australia for the comments that help improved the paper.

References

1. Culler, D.E., Estrin, D., Srivastava, M.B.: Guest editors' introduction: Overview of sensor networks. *IEEE Computer* 37, 41–49 (2004)
2. Chu, D.C., Popa, L., Tavakoli, A., Hellerstein, J.M., Levis, P., Shenker, S., Stoica, I.: The design and implementation of a declarative sensor network system. In: *The 5th ACM Conference on Embedded Networked Sensor Systems (SenSys 2007)*, Sydney, Australia, pp. 175–188 (November 2007)
3. Madden, S., Franklin, M.J., Hellerstein, J.M., Hong, W.: The design of an acquisitional query processor for sensor networks. In: *SIGMOD Conference*, pp. 491–502 (2003)
4. Yao, Y., Gehrke, J.: The Cougar approach to in-network query processing in sensor networks. *ACM SIGMOD Record* 31, 9–18 (2002)
5. Gehrke, J., Madden, S.: Query processing in sensor networks. *Pervasive Computer* 3, 46–55 (2004)
6. Considine, J., Li, F., Kollios, G., Byers, J.W.: Approximate aggregation techniques for sensor databases. In: *Proceedings of the 20th International Conference on Data Engineering, ICDE 2004*, pp. 449–460. IEEE Computer Society, Los Alamitos (2004)
7. Lloyd, J.W.: *Foundations of Logic Programming*, 2nd edn. Springer, Heidelberg (1987)
8. Leuschel, M.: Logic program specialisation. In: *Partial Evaluation*, pp. 155–188 (1998)
9. Compton, M.: Finding equivalent rewritings with exact views. In: *Proceedings of the 25th International Conference on Data Engineering, ICDE 2009*, pp. 1243–1246. IEEE, Los Alamitos (2009)
10. Li, L., Taylor, K.: A framework for semantic sensor network services. In: Bouguet-taya, A., Krueger, I., Margaria, T. (eds.) *ICSOC 2008*. LNCS, vol. 5364, pp. 347–361. Springer, Heidelberg (2008)
11. Sterling, L., Shapiro, E.: *The art of Prolog: advanced programming techniques*. MIT Press, Cambridge (1986)
12. Lloyd, J.W., Shepherdson, J.C.: Partial evaluation in logic programming. *J. Log. Program.* 11, 217–242 (1991)
13. Lakhotia, A., Sterling, L.: How to control unfolding when specializing interpreters. *New Generation Comput.* 8, 61–70 (1990)
14. Deusch, A., Ludäscher, B., Nash, A.: Rewriting queries using views with access patterns under integrity constraints. *Theor. Comput. Sci.* 371, 200–226 (2007)
15. Hellerstein, J.M., Hong, W., Madden, S., Stanek, K.: Beyond average: Toward sophisticated sensing with queries. In: Zhao, F., Guibas, L.J. (eds.) *IPSN 2003*. LNCS, vol. 2634, pp. 63–79. Springer, Heidelberg (2003)
16. Madden, S., Franklin, M.J., Hellerstein, J.M., Hong, W.: TinyDB: An acquisitional query processing system for sensor networks. *Transactions on Database Systems (TODS)* 30, 122–173 (2005)
17. Yao, Y., Gehrke, J.: Query processing in sensor networks. In: *Proceedings the 1st Biennial Conference Innovative Data Systems Research (CIDR)*. ACM Press, New York (2003)
18. Tavakoli, A., Chu, D., Hellerstein, J., Levis, P., Shenker, S.: A declarative sensor network architecture. In: *International Workshop on Wireless Sensor Network Architecture (WWSNA 2007)*, Cambridge, Massachusetts (April 2007)
19. Galpin, I., Brenninkmeijer, C.Y.A., Jabeen, F., Fernandes, A.A.A., Paton, N.W.: An architecture for query optimization in sensor networks. In: *Proceedings of the 24th International Conference on Data Engineering, ICDE 2008*, pp. 1439–1441. IEEE, Los Alamitos (April 2008)
20. Delin, K., Jackson, S.: The sensor web: a new instrument concept. In: *Proceedings of the SPIE International of Optical Engineering*, vol. 4284, pp. 1–9 (2001)

Conditional Localization and Mapping Using Stereo Camera

Jigang Liu, Maylor Karhang Leung, and Daming Shi

School of Computer Engineering
Nanyang Technological University
BLK. N4, Nanyang Avenue, Singapore 639798
School of Electrical Engineering and Computer Science
Kyungpook National University
1370, Sankyuk-dong, Buk-gu, Daegu, 702-701, Korea
liuj0028@ntu.edu.sg, asmkleung@ntu.edu.sg, dmshi@ee.knu.ac.kr

Abstract. In this paper, conditional localization and mapping (CLAM) is realized with a stereo camera as the only sensor. Compared with visual simultaneous localization and mapping (SLAM), the framework of CLAM is a novel proposed conditional filter rather than extended Kalman filter (EKF). In this algorithm, there is no camera velocity information in the filter state, the measurements and state equation all depend on image data which are the most reliable information so that CLAM outperforms SLAM when the camera turns abruptly or there are some frames lost in which conditions the SLAM may diverge quickly because the predefined model is incorrect in such cases. For CLAM, the model is derived from image data so that CLAM has no such problems. The experimental results show that the proposed CLAM is robust to abrupt turning of the camera and frame-losing, and also give the precise 3D information about the features and the trajectory of the camera.

Keywords: Stereo Camera, CLAM, conditional filter.

1 Introduction

In the past decade, significant progress has been made in autonomous robot navigation. SLAM has become more and more popular in robotics as a solution to the question of a moving sensor platform constructing a map of its environment during its first navigation while concurrently estimating its position and direction [1, 2, 3].

Early work was done in sonar-based navigation of mobile robots using the Kalman filter algorithm, as in [4] and [5]. Although sonar signals are insensitive to illumination variance, they are inaccurate. Compared with sonar signals, images captured by camera are compact, accurate, and well understood.

As for visual map building, Moutarlier and Chatila [6] proposed an approach taking account of all correlations in general robot localization and mapping problems within a single state vector and covariance matrix updated by the extended Kalman filter (EKF). Several early implementations verified the single EKF approach for

building modest-sized maps in real robot systems and convincingly demonstrated the importance of maintaining estimate correlations. These successes gradually resulted in very widespread adoption of EKF as the core estimation technique in SLAM. The most successful visual SLAM using a monocular camera was recently developed by Davision [7, 8, 9], whose contributions include an active approach to mapping and measurement, the use of a general motion model for smooth camera movement, and solutions for monocular feature initialization and feature orientation estimation. Civera [10, 11, 12] enhanced Davision's work by introducing inverse depth for feature points, producing measurement equations with a high degree of linearity. Thomas [13, 14] realized vision-based SLAM using stereo camera, monocular camera and Panoramic camera, respectively. This approach can deal with close large loops. All the above approaches built a map of the environment with feature points and the trajectory of the camera. However, many images must be obtained in a short time. In addition, the camera should move smoothly because for EKF, if the estimate of the state is wrong, EKF may diverge quickly owing to its linearization.

In this paper, we address the problem that the filter diverges when the camera turns abruptly. We are inspired by the conditional filters [15] first proposed for point tracking. The proposed CLAM also includes a condition with respect to image data. The camera state is predicted from image data which is much more reliable information than the previous knowledge. In the case of monocular SLAM [9], it faces the problem of the scale so that it needs additional sensors or some a priori knowledge. Stereo camera that provides scale through the baseline is used in the proposed CLAM. For the close points which present large disparity on the stereo image, they are initialized as 3D points which will provide distance and orientation information.

The paper is organized as follows. In Section 2, we introduce the conditional filter. Section 3 gives the details of the CLAM system. Section 4 provides the experimental results. In Section 5, we draw the conclusion of this paper and future work.

2 Conditional Filter

In [15], a conditional linear filter and a conditional nonlinear filter are derived based on Kalman filter and particle filters, respectively. For our cases, we propose another conditional filter as an extension of EKF with respect to image data.

Let \mathbf{I}_k denote an image obtained at time k . The sequence of images $\{\mathbf{I}_k, k = 0, \dots, n\}$ will be represented by $\mathbf{I}_{0:n}$. The nonlinear image-based filtering problem can be represented by the following system:

$$\mathbf{x}_k = f_k^{\mathbf{I}_{0:k}}(\mathbf{x}_{k-1}, \mathbf{w}_k^{\mathbf{I}_{0:k}}) \quad (1)$$

$$\mathbf{z}_k = h_k^{\mathbf{I}_{0:k}}(\mathbf{x}_{k-1}, \mathbf{v}_k^{\mathbf{I}_{0:k}}) \quad (2)$$

The index $\mathbf{I}_{0:k}$ indicates a dependence on the image data. Note that \mathbf{x}_k is the system state and \mathbf{z}_k is the measurement at time k . Functions $f_k^{\mathbf{I}_{0:k}}$ and $h_k^{\mathbf{I}_{0:k}}$ may be estimated

from \mathbf{I}_{0k} . Variables \mathbf{w}_k^{10k} and \mathbf{v}_k^{10k} , which are process noise and measurement noise, respectively are zero mean independent white noises with covariances \mathbf{Q}_k^{10k} and \mathbf{R}_k^{10k} respectively.

A condition is included with respect to image data; thus the equations for the optimal filter can be applied to the proposed model.

The state is assumed to be a Gaussian vector. The Gaussian probability density function (pdf) is completely characterized by the mean and covariance matrix. The filter can be represented by a recursive process including prediction and update phases. The process goes like this:

Step 1. Initialization of $\hat{\mathbf{x}}_0, \mathbf{P}_0$.

where $\hat{\mathbf{x}}_0$ is the filter state, \mathbf{P}_0 is the associated state covariance.

Step 2. Estimation of functions and matrices $f_k^{10k}, h_k^{10k}, \mathbf{Q}_k^{10k}$ and \mathbf{R}_k^{10k} from the image sequence.

Step 3. Prediction

$$\begin{aligned} \hat{\mathbf{x}}_{k|k-1} &= f_k^{10k}(\hat{\mathbf{x}}_{k-1}) \\ \mathbf{P}_{k|k-1} &= \mathbf{F}_k^{10k} \mathbf{P}_{k-1} (\mathbf{F}_k^{10k})^T + \mathbf{Q}_k^{10k} \end{aligned}$$

$$\text{where } \mathbf{F}_k^{10k} = \left. \frac{\partial f_k^{10k}}{\partial \mathbf{x}} \right|_{\mathbf{x}_{k-1}}$$

Step 4. Compute \mathbf{z}_k

Step 5. Update:

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + \mathbf{R}_k)^{-1}$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k (\mathbf{z}_k - \hat{\mathbf{z}}_k)$$

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k) \mathbf{P}_{k|k-1}$$

$$\text{where } \mathbf{H}_k = \left. \frac{\partial h_k^{10k}}{\partial \mathbf{x}} \right|_{\mathbf{x}_{k-1}}$$

Step 6. Repeat Steps 2-5 until no unprocessed image remains.

$\hat{\mathbf{z}}_k$ is the predicted measurement derived from $\hat{\mathbf{x}}_{k|k-1}$ according to

$$\hat{\mathbf{z}}_k = h_k^{10k}(\hat{\mathbf{x}}_{k|k-1}) \tag{3}$$

\mathbf{z}_k is the measurement obtained from image \mathbf{I}_k . It is always computed by matching technique.

3 CLAM System

The difference between the SLAM and the CLAM is shown in Figure 1. Here, we take one loop from the two recursive processes for example.

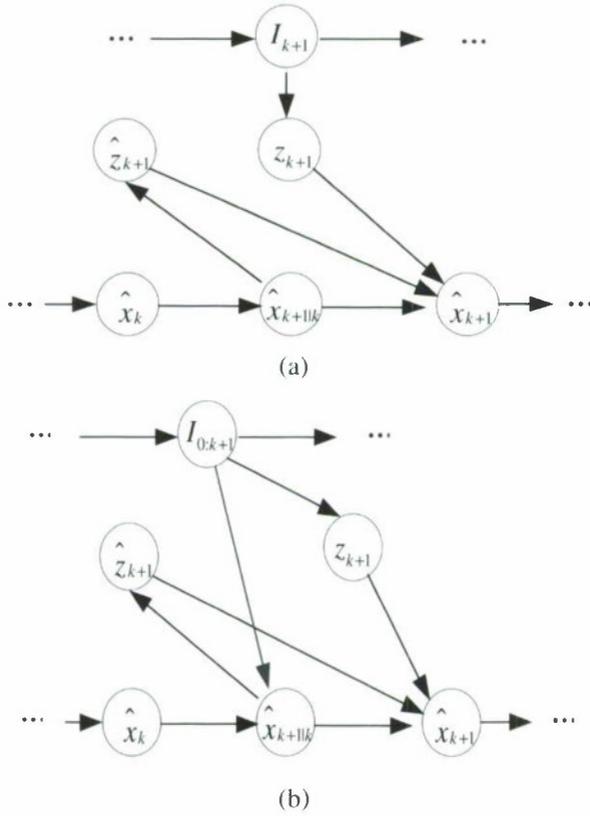


Fig. 1. The difference between the SLAM and the CLAM. (a) The process of the SLAM (b) the process of the CLAM.

For the SLAM, $\hat{x}_{k+1|k}$ is derived from \hat{x}_k according to the linear and angle velocity of the camera that may not be precise in certain conditions such as abrupt turning, frame losing, actually the velocity that is used to predict the state in time $k + 1$ is the velocity of the camera in time k . While in the CLAM, $\hat{x}_{k+1|k}$ is computed from \hat{x}_k and $\mathbf{I}_{k:k+1}$, the change of speed and direction of the camera, is computed from \mathbf{I}_k and \mathbf{I}_{k+1} .

3.1 The State Vector

The state of the CLAM \mathbf{x} is composed of the camera state \mathbf{x}_c and feature states $\mathbf{x}_{F_{L_n}}$. As a matter of fact, the result is represented by all the states of the CLAM system

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_c \\ \mathbf{x}_{F_{L_n}} \end{bmatrix} \tag{4}$$

The associated state covariance

$$\mathbf{P} = \begin{bmatrix} \mathbf{P}_{\mathbf{x}_c, \mathbf{x}_c} & \mathbf{P}_{\mathbf{x}_c, \mathbf{x}_{f/in}} \\ \mathbf{P}_{\mathbf{x}_{f/in}, \mathbf{x}_c} & \mathbf{P}_{\mathbf{x}_{f/in}, \mathbf{x}_{f/in}} \end{bmatrix} \quad (5)$$

The stereo camera is described by the position of its optical center \mathbf{r}^w and its orientation in Euler angles $\boldsymbol{\varphi}^w$

$$\mathbf{x}_c = \begin{pmatrix} \mathbf{r}^w \\ \boldsymbol{\varphi}^w \end{pmatrix} \quad (6)$$

For feature states, Inverse depth [10] used in monocular SLAM is proved to represent the distribution of features at infinity as well as close points, allowing performing an undelayed initialization of features. Despite its properties, each inverse depth point needs an over-parameterization of six values instead of a simpler three coordinate spatial representation. This produces a computational overhead. Here, working with a stereo camera, which can estimate the depths of points, the feature point is defined in terms of Euclidean coordinates.

$$\mathbf{x}_f = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \frac{b(u_l - u_0)}{u_l - u_r} \\ \frac{b(v_r - v_0)}{u_l - u_r} \\ \frac{fb}{u_l - u_r} \end{bmatrix} \quad (7)$$

where b is the baseline of the stereo camera, f is the focal length of the camera, (u_0, v_0) is the image center, (u_l, v_l) and (u_r, v_r) are the image coordinates on the left and right images respectively.

3.2 Prediction of the State

In this step, the state is estimated from current and previous images.

Firstly, for both left and right images, the image coordinate of the feature point in time k is estimated from $\mathbf{I}_{k-1,k}$ according to the robust parametric motion estimation approach [16]

$$(u_{lk}, v_{lk})^T = (u_{l(k-1)}, v_{l(k-1)})^T + B((u_{l(k-1)}, v_{l(k-1)})^T) \boldsymbol{\theta}_{lk} + \boldsymbol{\omega}_{lk} \quad (8)$$

$$(u_{rk}, v_{rk})^T = (u_{r(k-1)}, v_{r(k-1)})^T + B((u_{r(k-1)}, v_{r(k-1)})^T) \boldsymbol{\theta}_{rk} + \boldsymbol{\omega}_{rk} \quad (9)$$

where $B((u, v)^T) = \begin{bmatrix} 1 & u & v & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & u & v \end{bmatrix}$, $(u_{lk}, v_{lk})^T$ and $(u_{rk}, v_{rk})^T$ are the coordinates of the same feature point on the left and right images at time k . $\boldsymbol{\theta}_k$ is the parameter

vector which contains the polynomial's coefficients. ω is assumed to be a white noise of zero mean and covariance with respect to image data.

Secondly, depending on the estimated position of feature points on the both images at time $k-1$ and k , the position and direction of the feature can be derived with respect to the camera

$$\mathbf{x}_{f_i(k-1)} = \begin{bmatrix} \frac{b(u_{r(k-1)} - u_0)}{u_{l(k-1)} - u_{r(k-1)}} & \frac{b(v_{r(k-1)} - v_0)}{u_{l(k-1)} - u_{r(k-1)}} & \frac{fb}{u_{l(k-1)} - u_{r(k-1)}} \end{bmatrix}^T \quad (10)$$

$$\mathbf{x}_{f_k} = \begin{bmatrix} \frac{b(u_{rk} - u_0)}{u_{lk} - u_{rk}} & \frac{b(v_{rk} - v_0)}{u_{lk} - u_{rk}} & \frac{fb}{u_{lk} - u_{rk}} \end{bmatrix}^T \quad (11)$$

Using the two sets of correspondent points, $\{\mathbf{x}_{f_i(k-1)}\}$ and $\{\mathbf{x}_{f_k}\}$ the translation and rotation of the camera can be computed using the method in [17]. By defining:

$$\begin{aligned} \overline{\mathbf{x}_{f_i(k-1)}} &= \frac{1}{N} \sum_{i=1}^N \mathbf{x}_{f_i(k-1)} & \mathbf{x}_{f_i(k-1)}^v &= \mathbf{x}_{f_i(k-1)} - \overline{\mathbf{x}_{f_i(k-1)}} \\ \overline{\mathbf{x}_{f_k}} &= \frac{1}{N} \sum_{i=1}^N \mathbf{x}_{f_k} & \mathbf{x}_{f_k}^v &= \mathbf{x}_{f_k} - \overline{\mathbf{x}_{f_k}} \end{aligned} \quad (12)$$

A correlation matrix \mathbf{H} is defined by:

$$\mathbf{H} = \sum_{i=1}^N \mathbf{x}_{f_k}^v (\mathbf{x}_{f_i(k-1)}^v)^T \quad (13)$$

The singular value decomposition of \mathbf{H} is given by $\mathbf{H} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^T$, the optical rotation matrix \mathbf{R} can be derived by $\mathbf{R} = \mathbf{V}\mathbf{U}^T$. The optical translation of the camera \mathbf{T} can be calculated by $\mathbf{T} = \overline{\mathbf{x}_{f_i(k-1)}} - \mathbf{R}\overline{\mathbf{x}_{f_k}}$.

$$\mathbf{x}_{i \in \{k-1\}} = \begin{bmatrix} \mathbf{r}_{k|k-1}^w \\ \boldsymbol{\phi}_{k|k-1}^w \end{bmatrix} = \begin{pmatrix} \mathbf{r}_{k-1}^w + \mathbf{T} \\ \boldsymbol{\phi}_{k-1}^w \times \mathbf{R} \end{pmatrix} \quad (14)$$

For the feature points, because they are static with respect to the 3D map, they are predicted as the same with the previous state.

3.3 Feature Point Selection and Management

A good feature point selection algorithm is important for the whole system. A point is considered to be tracked reliably if its neighborhood defines a luminance pattern that carries enough information. To discard areas with insufficient luminance gradient, we use the selection criterion proposed in [18] (see Figure 2). This criterion is based on the eigenvalues of the structure tensor T

$$T((u, v)^T) = \int_{\mathfrak{R}((u, v)^T)} \begin{bmatrix} \nabla \mathbf{I}_u^2 & \nabla \mathbf{I}_v \nabla \mathbf{I}_u \\ \nabla \mathbf{I}_u \nabla \mathbf{I}_v & \nabla \mathbf{I}_v^2 \end{bmatrix} \quad (15)$$

where $[\nabla \mathbf{I}_u, \nabla \mathbf{I}_v] = [\partial \mathbf{I}_0 / \partial u, \partial \mathbf{I}_0 / \partial v]$, the two eigenvalues λ_1 and λ_2 give information on the intensity profile within $\mathfrak{R}((u, v)^T)$. Small eigenvalues are associated with a constant intensity profile, whereas large values indicate a luminance pattern that can be successfully tracked. The corresponding feature is therefore accepted as $\min(\lambda_1, \lambda_2) > \lambda$. λ is a threshold. The corresponding 11×11 patch with the feature point as its center is stored for measurement detection.



Fig. 2. Feature points detected using a stereo camera

At each step, we use only those features that fall in the field of view of both the left and right camera. Then project these features on the right and left images. A matching search based on normalized cross-correlation is performed using the patch associated with each feature. When insufficient features are visible, new features are added into the state. Moreover, non-persistent features are deleted from the state vector to avoid an unnecessary growth of the feature population.

When a new feature \mathbf{x}_f is added into the state

$$\mathbf{x} = \begin{bmatrix} \mathbf{X}_c \\ \mathbf{X}_{F_{1:n}} \\ \mathbf{X}_f \end{bmatrix} \quad \mathbf{P} = \begin{bmatrix} \mathbf{P}_{\mathbf{x}_c, \mathbf{x}_c} & \mathbf{P}_{\mathbf{x}_c, \mathbf{x}_{F_{1:n}}} & \mathbf{0} \\ \mathbf{P}_{\mathbf{x}_{F_{1:n}}, \mathbf{x}_c} & \mathbf{P}_{\mathbf{x}_{F_{1:n}}, \mathbf{x}_{F_{1:n}}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{P}_{\mathbf{x}_f, \mathbf{x}_f} \end{bmatrix} \quad (16)$$

When a feature \mathbf{x}_f is deleted from the state

$$\mathbf{x} = \begin{bmatrix} \mathbf{X}_c \\ \mathbf{X}_{F_{1:n}} \\ \mathbf{X}_f \end{bmatrix} \quad \mathbf{P} = \begin{bmatrix} \mathbf{P}_{\mathbf{x}_c, \mathbf{x}_c} & \mathbf{P}_{\mathbf{x}_c, \mathbf{x}_{F_{1:n}}} & \mathbf{P}_{\mathbf{x}_c, \mathbf{x}_f} \\ \mathbf{P}_{\mathbf{x}_{F_{1:n}}, \mathbf{x}_c} & \mathbf{P}_{\mathbf{x}_{F_{1:n}}, \mathbf{x}_{F_{1:n}}} & \mathbf{P}_{\mathbf{x}_{F_{1:n}}, \mathbf{x}_f} \\ \mathbf{P}_{\mathbf{x}_f, \mathbf{x}_c} & \mathbf{P}_{\mathbf{x}_f, \mathbf{x}_{F_{1:n}}} & \mathbf{P}_{\mathbf{x}_f, \mathbf{x}_f} \end{bmatrix} \quad (17)$$

3.4 Measurement Equation

At each step, we project every 3D feature point on the left image. A match is detected after performing normalized cross-correlation. In the following, a new measurement \mathbf{z} is used to update the state of the filter.

First, using the estimates of camera position $\mathbf{r}_{k|k-1}^w$ and feature position \mathbf{x}_{f_i} , the position of the feature point relative to the camera $\mathbf{x}_{f_i}^R$ is expected to be:

$$\mathbf{x}_{f_i}^R = \begin{pmatrix} \mathbf{x}_{f_i,x}^R \\ \mathbf{x}_{f_i,y}^R \\ \mathbf{x}_{f_i,z}^R \end{pmatrix} = \mathbf{R}^w (\mathbf{x}_{f_i} - (\mathbf{r}_{k|k-1}^w - [\frac{b}{2}, 0, 0]^T)) \tag{18}$$

where \mathbf{R}^w is the rotation matrix. $\mathbf{r}_{k|k-1}^w - [\frac{b}{2}, 0, 0]^T$ is the position of the left camera.

The position at which the feature point \mathbf{x}_{f_i} would be found in the left image is calculated according to the standard pinhole model:

$$\mathbf{z}_i = \begin{pmatrix} u_i \\ v_i \end{pmatrix} = \begin{pmatrix} u_0 - fs_u \frac{\mathbf{x}_{f_i,x}^R}{\mathbf{x}_{f_i,z}^R} \\ v_0 - fs_v \frac{\mathbf{x}_{f_i,y}^R}{\mathbf{x}_{f_i,z}^R} \end{pmatrix} \tag{19}$$

where f is the focal length of the camera, (u_0, v_0) is the principal point, s_u and s_v are the camera calibration parameters.

4 Experiments

In order to demonstrate the robustness of the proposed CLAM system, we captured one short video with frame rate 20fps by a stereo camera Bumblebee2 (See Figure 3)



Fig. 3. Bumblebee2 stereo camera

The stereo camera provides a 100×84 degree FOV per camera, and has a baseline of 12cm. Features are initialized as 3D points which are less than 7m far away from the camera. With the help of the Tricops SDK supplied with the stereo camera, the

derived video is rectified so that in the CLAM system, the effect of distortion of lens is not considered.

The video is captured with the camera in hand. It is processed in Matlab with the proposed algorithm on a laptop with an Intel 4 processor at 1.8 GHz and 1G memory.

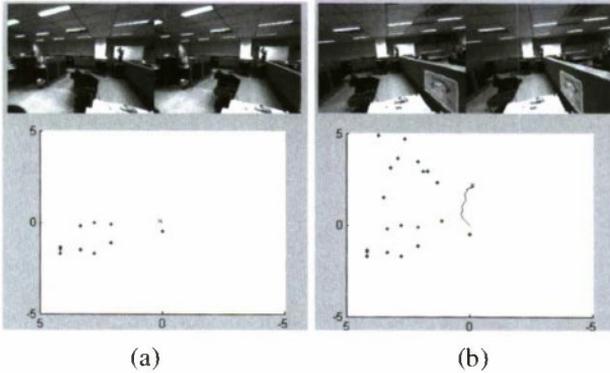


Fig. 4. Two frames of the video clip and the results of the CLAM (a) Frame #2 (b) Frame # 122

Figure 4 illustrates the results of the proposed CLAM. In order to show the CLAM is robust to frame losing and abrupt turning of camera, in the following experiment, frames from #20 to #40 are not used in the CLAM, which means that # 41 is to be processed after #19 is processed. The results are shown in Figure 5. The CLAM can also give correct estimate even when some frames are lost.

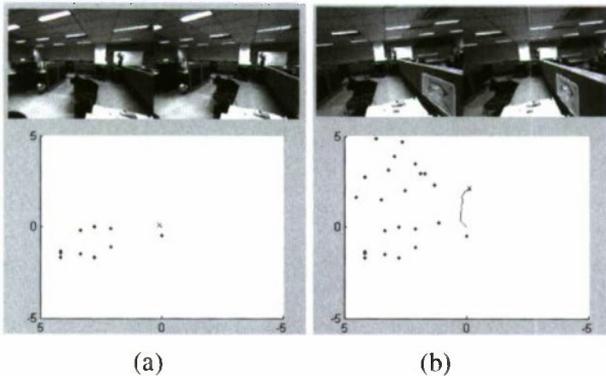


Fig. 5. Results of the CLAM in the case of frame losing (a) Frame #2 (b) Frame #122

In order to compare CLAM using stereo camera and SLAM using monocular camera, we take the all the left images from the video clip for implementing SLAM and at the first step, the stereo images are used to supply the SLAM with scale information. Table 1 compares the results of the camera's position using two

methods—CLAM with stereo camera and monocular SLAM. From the results, we can see that CLAM performs better; its trajectory is closed to ground truth, especially during an abrupt turn or frame-losing. For the SLAM, it deviated from the ground truth when it processed frame # 41 due to the incorrect velocity information of the camera used.

Table 1. The comparison of the camera's position calculated by CLAM and SLAM in 3 steps

Frame	Ground Truth (m)			Estimated (m)			
	X	Y	Z		X	Y	Z
#1	0.00	0.00	0.00	CLAM	0.00 ± 0.01	0.00 ± 0.01	0.00 ± 0.01
				SLAM	0.00 ± 0.01	0.00 ± 0.01	0.00 ± 0.01
#19	0.51	0.04	0.50	CLAM	0.50 ± 0.02	0.03 ± 0.02	0.50 ± 0.02
				SLAM	0.52 ± 0.03	0.04 ± 0.03	0.50 ± 0.02
#41	1.41	0.02	0.04	CLAM	1.40 ± 0.02	0.02 ± 0.03	0.04 ± 0.02
				SLAM	0.53 ± 0.5	0.03 ± 0.4	0.51 ± 0.5

5 Conclusions and Future Work

A significant contribution of this paper is to introduce a new localization and mapping method called CLAM. Compared with the traditional SLAM approach, the framework of the CLAM is an image-sequence-based conditional filter robust to occlusion and abrupt changes. According to robust parametric estimation technique, the velocity of the feature of interest can be derived to estimate the motion of the camera. Normalized cross-correlation is used to find the measurement.

In this research, a stereo camera is used as the only sensor, the nearby features are easy to initialize and provide the scale information to the 3D map. The close features provides distance and orientation information.

Currently, the CLAM is applied in a short video. In the future, we will implement the CLAM over a long distance, and improve it so that it is robust for loop detection.

As an extension of SLAM, CLAM currently is focused on building 3D maps of unknown environments with feature points, which can be easily detected and identified, but not robust against occlusion and illumination. In indoor environments, many line features are available, such as the edges of walls, tables, etc. Lines have various advantages over points. First, lines are insensitive to illumination and occlusion. Second, maps of line segments visualize the spatial structures of environments. Third, line matching can be achieved even when viewpoint changes occur, but point features can only be reliably matched over a narrow range of viewpoints. In the future, we will target at building a map with line segments!

References

1. Durrant-Whyte, H., Bailey, T.: Simultaneous localization and mapping (SLAM): Part I the essential algorithm. *IEEE Transaction on Robotics and Automation* 13(2), 99–110 (2006)
2. Bailey, T., Durrant-Whyte, H.: Simultaneous localization and mapping (SLAM): Part II. *IEEE Robotics and Automation Magazine* 13(3), 108–117 (2006)
3. Dissanayake, M.W.M.G., Newman, P., Clark, S., Durrant-Whyte, H.F., Csorba, M.: A solution to the simultaneous localization and map building (SLAM) problem. *IEEE Transactions on Robotics and Automation* 17(3), 229–241 (2001)
4. Crowley, J.: World modeling and position estimation for a mobile robot using ultra-sonic ranging. In: *IEEE International Conference on Robotic and Automation*, pp. 674–680 (1989)
5. Laumond, J.P., Chatila, R.: Position referencing and consistent world modeling for mobile robots. In: *IEEE International Conference on Robotic and Automation* (1985)
6. Moutarlier, R., Chatila, R.: Stochastic Multisensory Data Fusion for Mobile Robot Location and Environment Modelling. In: *Proceeding International Symp. on Robotics Research* (1989)
7. Davison, A.J., Kita, N.: 3D simultaneous localisation and map-building using active vision for a robot moving on undulating terrain. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2001)
8. Davison, A.J., Murray, D.W.: Simultaneous localization and map-building using active vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24(7), 865–880 (2002)
9. Davison, A.J., Reid, I.D., Molton, N.D., Stasse, O.: MonoSLAM: real-time single camera SLAM. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29(6), 1052–1067 (2007)
10. Civera, J., Davison, A.J., Montiel, J.M.M.: Inverse depth to depth conversion for monocular SLAM. In: *IEEE International Conference on Robotics and Automation* (2007)
11. Davison, A.J., Civera, J., Montiel, J.M.M.: Inverse Depth Parametrization for Monocular SLAM. *IEEE Transaction on Robotics* (2008)
12. Montiel, J.M.M., Civera, J., Davison, A.J.: Unified inverse depth parametrization for monocular SLAM. In: *Proceedings of Robotics: Science and Systems* (2006)
13. Thomas, L., Lacroix, S.: SLAM with panoramic vision. *Journal of Field Robotics* 24(1-2), 91–111 (2007)
14. Thomas, L., Cyrille, B., Il-Kyun, J., Simon, L.: Vision-based SLAM: stereo and monocular approaches. *International Journal of Computer Vision* 74(3), 343–364 (2007)
15. Arnaud, E., Memin, E., Cernusechi-Frias, B.: Conditional filters for image sequence-based tracking - application to point tracking. *IEEE Transactions on Image Processing* 14(1), 63–79 (2005)
16. Odobez, J.M., Bouthemy, P.: Robust multiresolution estimation of parametric motion models. *Journal of Visual Communication and Image Representation* 6(4), 348–365 (1995)
17. Eggert, D.W., Lorusso, A., Fisher, R.B.: Estimating 3-D rigid body transformations: a comparison of four major algorithms. *Machine Vision and Application* 9(5-6), 272–290 (1997)
18. Tomasi, C., Shi, J.B.: Good features to track. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (1994)

A Unified Approach for Extracting Multiple News Attributes from News Pages

Wei Liu^{1,2}, Hualiang Yan¹, Jianwu Yang^{1,2}, and Jianguo Xiao¹

¹Institute of Computer Science & Technology, Peking University

²Key Laboratory of Computational Linguistics (Peking University), MOE
China, 100871

{liuwei, yanhualiang, yangjianwu, xjg}@icst.pku.edu.cn

Abstract. Most previous works on web news article extraction only focus on its content and title. To meet the growing demand for the various web data integration applications, more useful news attributes, such as publication date, author, etc., need to be extracted structured stored for further processing. In this paper, we study the problem of automatically extracting multiple news attributes from news pages. Unlike the traditional ways (e.g. extracting news attributes separately or generating template-dependent wrappers), we propose an automatic, unified approach to extract them based on the visual features of news attributes which includes independent visual features and dependent visual features. The basic idea of our approach is that, first, the candidates of each news attribute are extracted from the news page based on their independent visual features, and then, the true value of each attribute is identified from the candidates based on dependent visual features (the layout relations among news attributes). The extensive experiments using a large number of news pages show that the proposed approach is highly effective and efficient.

Keywords: web data extraction, news attribute, visual feature.

1 Introduction

As one of the most popular web information sources, web news articles attract countless surfers every day. Meanwhile, many important applications need an efficient way to extract news articles from web pages at fine granularity instead of indexing the whole pages. Fig.1 shows an example of news article, and the attributes of this article are also marked with red boxes.

Extracting news articles from web pages automatically is always a very challenging task due to various layouts or templates of news web pages. To the best of our knowledge, though some efforts [1,2,13] have been done on this task, most of them only focus on extracting *content* and *title*. In fact, more attributes (*publication date*, *author*, etc.) also need to be extracted to meet to the growing demand of the various applications. Table 1 illustrates the functions of the news attributes except *title* and *content* (because their functions are widely known).

In this paper we focus on 8 important news attributes: *title*, *author*, *publication date*, *content*, *category*, *source*, *related news links*, *comment link*. Though *title* and *content* can be extracted with good performance using appropriate features (e.g. Html tag, font size, text length, etc.) in previous works [1, 13], most of attributes cannot be extracted in the similar way. For example, *publication date* would be difficult to be identified only with its own features if many dates appear in the news page. However, a user can still identify it without any difficulty according to the layout relations of it and other attributes.

In fact, when people browse a web page, they are subconsciously guided by the experience they have accumulated in browsing similar web pages. Therefore, to ease users' consumption, the news page designers always give a careful consideration on the visual features of news attributes, i.e., what type of font should be used and where it should be placed in the page. Our approach simulates how a user understands news attributes in news pages based on his visual perception. In this paper, the visual features of news attributes used in our approach are classified into two types: independent visual features and dependent visual features. Independent visual features are used to identify news attributes independently, including font, text length, etc. Dependent visual features characterize the layout relations among news attributes on web pages, including direction feature and neighbor feature. Section 2 will introduce these visual features.

Based on the visual features, we propose a new unified approach to extract news attributes, which is different to traditional ways which extract each attribute independently or generate template-dependent wrappers. The proposed approach consists of two stages. First, several candidates of each attribute are extracted from the news page based on their *independent visual features*. Next, the true value of each attribute is identified from its candidates based on the *dependent visual features*. A prototype system VEWNO has been implemented based on the proposed approach. Though 8 attributes are focused in this paper, our approach is general for the extraction task of any attribute set.

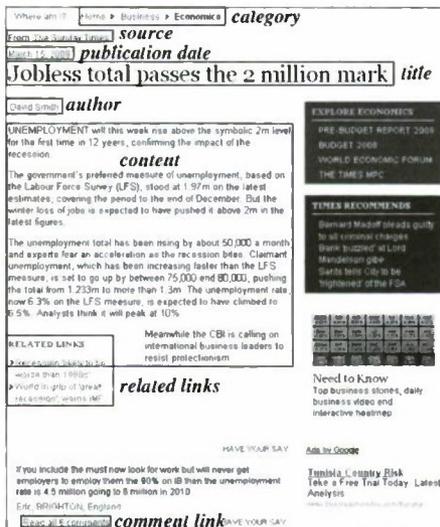


Fig. 1. An example of web news article

Table 1. The functions of news attributes

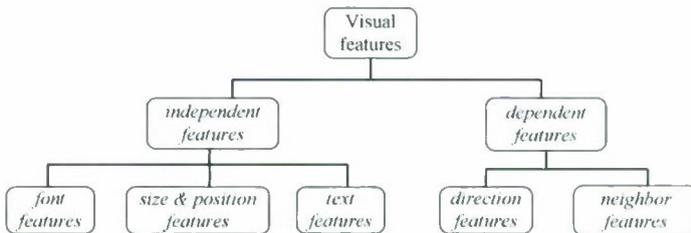
News attribute	Function
<i>category</i>	classifying news article on topic
<i>related links</i>	crawling event-oriented news articles
<i>comment link</i>	crawling comment pages related to this news article
<i>publication date</i>	ordering the news articles about the same event according to the time
<i>author, source</i>	evaluating the credibility of a news article

Overall, the contributions in this paper are summarized on four aspects. First, we investigate the visual features of news attributes (especially *dependent visual features*) of news attributes. To the best of our knowledge, the existing approaches mainly focused on some of *independent visual features*. Second, we propose a unified approach to extract multiple attributes of web news articles. Intuitively, extracting more attributes will bring more challenges. But we think such challenges are also opportunities because more informative evidences can be used to improve the extraction performance. In other words, the extraction performance of an attribute can be improved by the layout relations of it and other attributes, and vice versa. Third, our approach is also the combination of web data extraction and annotation. That is, the attributes have been assigned the right semantics when they are extracted. It is widely known that web data annotation is a very challenging task[12]. Fourth, the basic idea of our approach is not limited to the extraction task of news articles. It is a promising way to extract multiple attributes of web object simultaneously from noise-rich web pages. Many structured web objects, such as Blog and product, can also be extracted in the same way. As the future work, our approach will be applied to more other web objects.

The rest of the paper is organized as follows. In section 2, we introduce the visual features used in our approach. Section 3 and section 4 discuss the underlying techniques, attribute candidate extraction and true value identification, of our approach respectively. In section 5, we present and analyze the experimental results. The related work is introduced in section 6, and section 7 is the summary.

2 Visual Features

Web pages are special documents being accessed through a web browser. The visual information are the most important clues to help people understand the semantic structures of web pages. As a result, news attributes can also be identified with

**Fig. 2.** The category of visual features

appropriate visual features. The visual features of news attributes include *independent features* and *dependent features*. Fig. 2 shows the category of the visual features.

Independent features: For each news attribute, some visual features are very useful to identify it in news pages. For example, *title* always uses a notable font compared to other texts on web pages. We call these features *independent features* in this paper. Three kinds of *independent features* are listed as follows.

- *Font features:* size, bold, style, color;
- *Size & position features:* height, width, coordinate(x, y)¹;
- *Text features:* text length, link text length, frequent words, expression format (such as date).

Further, more advanced features can be derived from the basic features above. For examples, the area of a text block can be calculated with its width and height, and ratio of link texts can be calculated according the text length and the link text length. We do not list all the features due to the limitation of paper length.

Dependent features: We import *dependent features* to represent the layout relations among news attributes on Web pages. According to our observations, the layout relations of new attributes are not in chaos though the templates of news pages are various. We classify such regularities into *direction feature* and *neighboring feature*.

Direction feature indicates the direction relation among attributes. Since a web page is two-dimensionally laid out, we use “top-down” and “left-right” to represent this feature. The direction relation of two attributes a_1 and a_2 is defined as below.

- Top-down: $a_i.y < a_j.y$;
- Left-right: $a_i.y = a_j.y$ and $a_i.x < a_j.x$.

These relations can be deduced easily with their coordinates. According to the definition, “top-down” relation takes precedence of “left-right” relation, and so it is impossible that a_i is both on top and on the left of a_j in one news article.

Neighbor feature represents the neighbor relations among attributes. For example, *author* and *content* are often neighbors on the page. a_1 and a_2 are defined to the neighbor relation iff no other attributes appear between them on the horizontal or vertical direction. Note that, noise texts does not influence the neighbor relation. For example, in Fig. 1, the *related news links* and the *comment link* are still regarded as being neighboring even some texts are inserted between them.

	<i>title</i>	<i>author</i>	<i>content</i>
<i>title</i>	/		{1,0,38.5%}	
<i>author</i>		/		
<i>content</i>			/	
.....				/

Fig. 3. The model of layout relation matrix

¹ The origin is the top-left corner of a web page, and (x, y) is the top-left corner of the text block.

We use **layout relation matrix** to represent the layout relations between any two attributes. Fig. 3 shows the model of the layout relation matrix. Each cell in it is denoted in form of triple $\{p_t, p_l, p_n\}$, where p_t, p_l and p_n are the probabilities for top-down, left-right and neighbor relations respectively. For example, the cell $\{1,0,38.5\}$ means the probabilities of *title* and *content* on the three layout relations are 1, 0 and 38.5%. The layout relation matrix can be produced using labeled news pages. We observe that the probabilities in the layout relation matrix will be convergent when the number of news pages is large enough.

3 Attribute Candidate Extraction

Attribute candidates extraction targets at extracting some text blocks from the news page as the candidates of each news attribute and assure the true value must be one of them. In our implementation, a news page is partitioned into a set of text blocks. Any text block holds a rectangular area on the page, and the visual information (font, coordinate, etc.) is attached the text blocks during this process. We adopt the VIPS algorithm [15] to build the visual block tree for a news page and collect the leaf nodes as the initial text blocks. To ensure attributes and text blocks are 1:1 mappings, we merge the text blocks as one block if they share the same font, are adjoining on the page and are not separated by recognized separators (such as "/") [7].

Table 2. The rules for attribute candidate extraction

attribute	Extraction rules	attribute	Extraction rules
<i>title</i>	$45px \geq \text{Font-size} \geq 15px$	<i>source</i>	$\text{Font-size} \leq 12px$
	Font-color: black or blue		Font-color: black, grey or brown
	$y < \text{page. height}/2$		Frequent-word: "from", etc.
	$y < \text{screen. height}$		$4 \leq \text{Text-length} \leq 25$
	$8 < \text{Text-length} < 50$		$6px \leq \text{font-size} \leq 12px$
<i>publication date</i>	isAnchorText: no	<i>content</i>	Font-color: black
	$\text{Font-size} \leq 10px$		$y < \text{screen. height}$
	Font-color: black, blue or grey		$\text{text-length} \geq 20$
	$\text{Text-length} \leq 16$		<i>category</i>
Text-format: date regular expressions	$y < \text{page. height}/2$		
isAnchorText: no	$y < \text{screen. height}$		
<i>author</i>	$\text{Font-size} \leq 12px$	<i>related news links</i>	$8 \leq \text{Text-length} \leq 30$
	$3 \leq \text{Text-length} \leq 25$		Frequent-word: ">", "→", "f", etc.
	Frequent-word: "author:", "By" etc.		$\text{Font-size} \leq 12px$
<i>comment link</i>	$\text{Font-size} \leq 12px$	<i>related news links</i>	Font-color: black or blue
	$6 \leq \text{Text-length} \leq 15$		$y > \text{page. height}/2$
	Frequent-word: "comment"		isAnchorText: yes
	isAnchorText: yes		Frequent-word: "related news", "related links", etc.

3.1 Candidate Extraction

For each attribute, some text blocks are extracted as its candidates using several simple heuristic rules based on independent features. The candidate extraction rules are already obtained, which are just the rules shown in Table 2. If a text block satisfies all the rules of some attribute, it will be regarded as one candidate of this attribute. In this way, a group of text blocks are extracted as the candidates for each attribute. Now we will introduce a general automatic way of training candidate extraction rules using labeled news pages.

3.2 Training Candidate Extraction Rules

To build these candidate extractors, two questions have to be answered: given a news attribute, which visual features are selected to generate candidate extraction rules and how to set appropriate values for the rules?

Visual feature selection

For each news attribute, some useful visual features are selected to generate candidate extraction rules. For example, font size is a very effective feature to help users detect *title* from a news page. Manually selecting visual features is time-consuming and error-prone. The classic algorithm C4.5 is employed for this task because it can select appropriate features and use them to build the classification tree. The training set is the text blocks obtained from news pages in the step of web page representation. In the training set, the true values are labeled as the positive samples are, and others are labeled as the negative samples. When the classification tree for each news attribute is built using C4.5 algorithm, the features in the classification tree are selected.

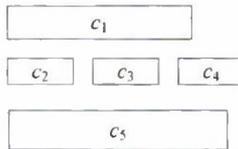


Fig. 4. An example to illustrate the neighbor relation

Candidate extraction rule generation

To assure the true value must satisfy every rules of the attribute, the value domain of a selected feature is the union of true values of this attribute in the training set on this feature. For example, the *title* candidate extraction rule “ $45\text{px} \geq \text{Font-size} \geq 15\text{px}$ ” means the range of *title*’s font size is from 15px to 45px in the training set. When the size of the training set is large enough, we believe the candidate extraction rules shown in Table 2 are safe.

4 True Value Identification

The goal of true value identification is to identify the true value from the candidates of each attribute. In this section, we first introduce the method of measuring the layout reasonableness of a candidate news article, and then propose an efficient way for true value identification. We define candidate news article to be the candidates from different attributes, and define true news article to be all of its candidates in it are true values. Obviously, true news article is more reasonable than other candidate news article on the layout.

4.1 Measuring the Layout Reasonableness of a Candidate New Article

We define the layout reasonableness of a candidate news article as the sum of the layout reasonableness of any two candidates in it. We measure the layout reasonableness of any two candidates based on the layout relation matrix. Given any two candidates c_i and c_j belonging to different news attributes, the layout reasonableness of them is calculated below:

$$\varphi(c_i, c_j) = \begin{cases} \lambda_{ij} \cdot p_x(a_i, a_j) + \mu_{ij} \cdot p_n(a_i, a_j) & i \neq j \\ 0 & \text{if } p_x(a_i, a_j) \cdot p_n(a_i, a_j) = 0 \end{cases} \quad (1)$$

where a_i and a_j are the attributes that the candidates c_i and c_j belong to, $p_x(a_i, a_j)$ and $p_n(a_i, a_j)$ are the probabilities of c_i and c_j on direction relation and neighbor relation respectively in the layout probability matrix. $p_x(a_i, a_j)$ is an alternative probability which is determined by the current direction relation of c_i and c_j ; if c_i and c_j satisfy the top-down relation, $p_x(a_i, a_j)$ is $p_t(a_i, a_j)$, otherwise is $p_l(a_i, a_j)$. λ_{ij} and μ_{ij} are the weights of p_x and p_n . For different attribute pair, the related λ and μ are also different. For instance, according to our observations, the direction relation is more important than the neighbor relation for *comment link* and *title*, while the neighbor relation of *comment link* and *title* is more important than the direction relation of them for *comment link* and

Algorithm of Calculating β
 Input: two candidates c_i and c_j
 Output: β
 Begin
 1 Initialize set C_x ;
 2 Put the candidate attributes that are between c_i and c_j into C_x ;
 3 For each $c_k \in C_x$ do
 4 $p(c_k) = \frac{\sum_{c_l \in CS_k} p(c_l, c_k)}{\sum_{l=1, l \neq k}^8 |CS_l|}$;
 5 $\beta = \prod_{c_k \in C_x} p(c_k)$
 6 Return β ;
 End

Fig. 5. Algorithm of calculating β

content. We choose SVM (Support Vector Machines) to obtain the appropriate values for λ_{ij} and μ_{ij} automatically. The training set is a set of candidate news articles. If all the attribute candidates in a candidate news article are true values, this candidate news article is labeled as the positive sample, else it is labeled as the negative sample.

However, it is difficult to determine whether two candidates are really neighboring when some other candidates are between them. Just like the example shown in Fig. 4, c_1 and c_5 are top-down relation, and three candidates (c_2, c_3, c_4) are between them. c_1 and c_5 are neighboring only when c_2, c_3 and c_4 are all false values, otherwise they are not. Because any candidate cannot be determined to be true or false yet, we use an attenuation factor β to represent the probability that two candidates are neighboring. Therefore Formula 1 is replaced by the following formula. Estimating β will be introduced soon.

$$\varphi(c_i, c_j) = \begin{cases} \lambda_{ij} \cdot p_x(a_i, a_j) + \mu_{ij} \cdot \beta \cdot p_n(a_i, a_j) & i \neq j \\ 0 & \text{if } p_x(a_i, a_j) \cdot p_n(a_i, a_j) = 0 \end{cases} \quad (2)$$

Further, the formula to measure the layout reasonableness of a candidate news article is given below.

$$\varphi(AC) = \begin{cases} \sum \varphi(c_i, c_j) & \\ 0 & \text{if any } \varphi(c_i, c_j) = 0 \quad c_i, c_j \in AC, i \neq j \end{cases} \quad (3)$$

Estimating β

β is the probability of c_i and c_j being neighboring in Formula 2, i.e., the probability that all candidates between c_i and c_j are false values. Suppose CS_k is the candidate set of attribute a_k that c_k belongs to, and its size is n . Intuitively, the probability that c_k is a false value, denoted as $p(c_k=F)$, should be $1-(1/n)$. But it is not reasonable. For example, *title* is always on top of *content*, so any candidate *title* must be a false value if it is under all candidate *contents*. Therefore, we propose a more effective algorithm (Fig. 5) to calculate β based on the direction relation.

Line 4 is an alternative component according to the direction relation of c_i and c_j . For example, if c_1 is on top of c_k , $p_x(c_1, c_k)$ is replaced by $p_t(c_1, c_k)$; if c_1 is on the left of c_k , $p_x(c_1, c_k)$ is replaced by $p_l(c_1, c_k)$. Obviously, the more candidates between c_i and c_j are, smaller β is.

4.2 Efficient Algorithm for True Value Identification

A straightforward way is to exhaust all possible candidate news articles and measuring the layout reasonableness of them. However, the total number of candidate news articles is often very large, i.e., $|CS_1| |CS_2| \dots |CS_8|$. Note that, if $|CS_i|=0$, it is removed from the formula. Because a large number of news pages are processed in real applications, it is inefficient to generate and measure all candidate news articles. To avoid such situation, a simple and efficient algorithm is proposed to find the *true news article*. The algorithm is an iterative process: first, all possible partial candidate news articles with only two candidate sets CS_1 and CS_2 are generated; next, when the candidate articles with $i-1$ candidate sets have been generated, the candidate news articles with i candidate sets are generated. The iterative process stops until $i=8$. At last, the optimal candidate article as the true news article is selected with Formula 2 from all candidate news articles. The details of this algorithm are shown in Fig. 6.

Algorithm for True News Article Identification**Input:** candidate sets of all attributes, CS_1, CS_2, \dots, CS_8 .**Output:** the true news article TNA .

Begin

```

1  Initialize  $ACS = \Phi$  //  $ACS$  is the article candidate set
//step 1: generate the partial article candidates with two candidate sets
2  For each  $c_i \in CS_1$  and each  $c_j \in CS_2$  do
3    Calculating  $\varphi(c_i, c_j)$  using Formula 2;
4    If  $\varphi(c_i, c_j) \neq 0$  then
5      Put  $\varphi(c_i, c_j)$  into  $ACS$ ;
//step 2: extend the current partial article candidates in  $ACS$  through adding
 $CS_3, \dots, CS_n$ 
6  For  $i=3$  to 8 do
7    For each  $AC_k \in ACS$  do
8      Take  $AC_k$  out of  $ACS$ ;
9      For each  $c_j \in CS_i$  do
10      $AC' = AC_k$ ;
11     Add  $c_j$  into  $AC'$ ;
12     If  $\varphi(AC') > \sigma$  then //  $\sigma$  is the threshold
13       Put  $AC'$  into  $ACS$ ;
//select and output the optimal article candidate
14   $TNA = \{AC \mid \varphi(AC) \text{ is max, } AC \in ACS\}$ ;
15  Return  $TNA$ ;
End

```

Fig. 6. Algorithm of true news article identification

There are two steps in this algorithm. In step 1 (lines 2-5), the partial candidate news articles (only containing two candidates) are generated with CS_1 and CS_2 , and the non-zero ones are putted into ACS . In step 2 (lines 6-13), when $i-1$ candidate sets have been processed, every candidate in CS_i is added into each partial candidate article in ACS . If the value of an extended partial candidate article is smaller than the predefined threshold σ , it will be removed from ACS because it has a low chance to be the optimal one. Such pruning operation can reduce the size of ACS greatly. At the end of the algorithm, the article candidate with the maximum value is selected as the optimal one, and the attribute candidates in it is regarded as the true values of news attributes.

5 Experiments

To evaluate the performance of our approach, we have implemented a prototype system VEWNO. The input is any web news page that is well-displayed in web browser, and the output is the news attributes embedded in this page. The current VEWNO can process one news page in 0.32 to 0.57 seconds (or 2-3 pages in one second).

5.1 Experiment Setup

The data set used in the experiments includes 50 online news sites. We randomly collect at least 100-300 news pages from every site. We divide the data set into the training

set and the test bed: the pages of 25 sites are as the training set, and the pages of the other 25 sites are as the test bed. The traditional measures, *precision*, *recall* and *F1*, are used in the experiments to evaluate VEWNO.

5.2 Performance Evaluation

We conduct the experiment to evaluate the performance of VEWNO. The training set is used to (1) learn the rules for attribute candidate extraction; (2) build the layout relation matrix; (3) train parameter λ_{ij} and μ_{ij} ; (4) estimate β .

Table 3. Extraction performance of VEWNO

	<i>title</i>	<i>author</i>	<i>source</i>	<i>publication date</i>
<i>precision</i>	98.2%	91.5%	94.9%	97.9%
<i>recall</i>	95.0%	84.3%	90.3%	96.1%
<i>F1</i>	96.2%	87.8%	92.5%	97.0%
	<i>content</i>	<i>review</i>	<i>category</i>	<i>related news links</i>
<i>precision</i>	96.4%	96.3%	97.7%	95.8%
<i>recall</i>	93.7%	93.1%	95.5%	92.2%
<i>F1</i>	95.0%	94.7%	96.6%	94.0%

Table 3 shows the experimental results of VEWNO on testing bed, and two conclusions can be made. First, the performance of approach is very good on both 8 attributes and three measures. Compared with the experimental results reported by [1] [5] on *content* and *title* respectively, our approach is better or very close to them. [13] is much better than ours because they only extract the minimum sub tree that contains *content* not the exact *content*. Second, our approach is template independent because the templates of the pages in training set are different from those of the pages in testing bed. So VEWNO can perform the news article extraction task to any news web pages. This trait is very important for real applications.

5.3 Experiments on Visual Features

To evaluate the effectiveness of the *dependent visual features*, we implemented 8 extractors which extract 8 news attributes from news pages separately and compare their performances with those of VEWNO. Each extractor is actually a classification tree trained by SVM(LibSVM) only based on *independent visual features*(IV), and VEWNO can be viewed as the combination of *independent visual features* and *dependent visual features*(DV).

The experimental results on *F1* measure show IV+DV outperforms only IV significantly on most attributes. For some attributes (*title*, *content* and *category*), the performances are acceptable only using the *independent visual features*. But for other attributes, the performances are very poor because their *independent visual features* are not distinguishable enough to other texts in the pages. So the experiment proves *dependent visual features* imported by us are very effective to improve the performance.

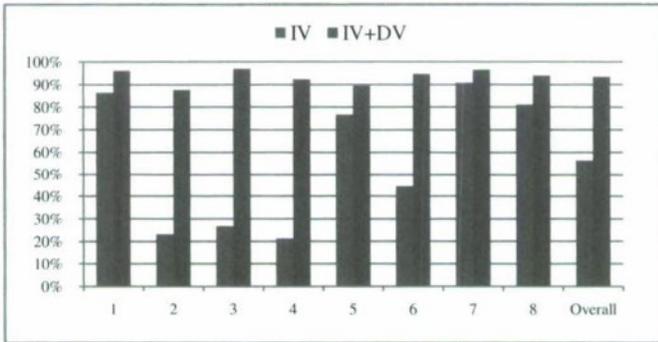


Fig. 7. Comparison experiments between IV and IV+DV

5.4 Experiments on Comparison with CRF-Based Approach

CRF is the state-of-art model for specific semantic object extraction. We have implemented the CRF-based approach proposed in [16] which is close to ours, and make the comparison. The training set and testing set are same to those of VEWNO.

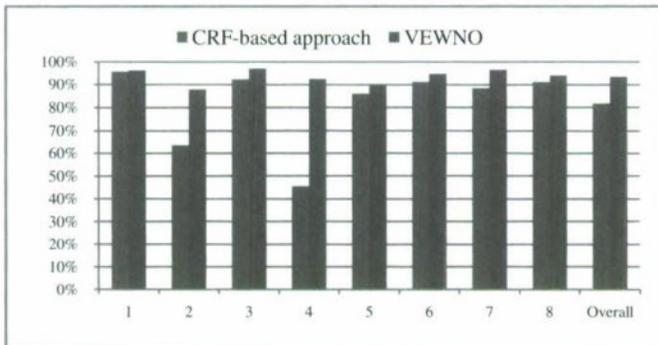


Fig. 8. Comparison experiments between VEWNO and CRF-approach

From the experimental results in Fig. 8, we find that, though the of CRF-based approach has better the performances of some attributes, it is still no match for VEWNO. The reason is that, though CRF-based approach also exploits the order dependencies among attributes, it overlooks the *neighbor feature* and its performance will be poor if too many false ones in the candidates. For example, the extraction performances of *author* and *source* are very poor because the direction features of them and *content* are not strong enough.

6 Related Work

The problem studied in this paper belongs to the field of web data extraction. It has received a lot of attentions in recent years. Survey [8] has given a good summary for these efforts. The research efforts in this field are either template-dependent [3,4,9,10,18] or template-independent[6,7,14,17]. In this section, we give a brief introduction for them first. Then, the works on news article extraction will be introduced and compared.

Template-dependent works mainly focus on extracting structured data records and data items in the web pages through inducing the common template. Most of them utilize the structure information on the DOM tree of a web page to represent the templates of similar web data records. In recent works, some visual features are also combined with the DOM tree to improve the performance, such as the method introduced by ViNTs[4]. However, the generated wrappers are sensitive by those works can only be applied for the web pages that share similar templates, and are not practical for the task of web news article extraction from general web sites. In addition, an annotation task is needed to assign right semantics for the extracted data. Template-independent works aim to extract structured data from different-template web pages. Most of these methods are based on probabilistic models, which integrate semantic information and human knowledge in inference. For example, Conditional Random Fields (CRF)[11] and its variations(such as 2D-CRF[6], HCRF[7] and Semi-Markov CRF[14]) infer the semantics of the text blocks in web pages by learning the order dependencies of web data distribution. The distinct advantage of them are insensitive to the templates of web pages. They are focusing on assigning the semantic label to the extracted data and can be seen as complementary to template-dependent works [6]. In addition, template-independent works are not suitable for the rich-noise web pages (such as news pages) because the too many noises will significantly weaken the dependency of web data distribution.

News extraction is a special topic in the field of web data extraction. Until now, several works have been proposed for web news article extraction, but most of them only focused on *content* extraction. [2] proposes a top-down approach to generate a tree-structured wrapper. This approach is template-dependent, so it is not practical when news pages come from different web sites. [1] is a template-independent approach for *content* extraction based on the *independent visual features*. But for most attributes, such as *publication date* and *source*, the performance will be poor if only their *independent features* are considered (see the experiments in section 6.3). Our approach is different to them on both application and technique. For application, our approach targets at extracting multiple news attributes not only *content*. For technique, our approach utilizes the layout relations of news attributes which can improve the extraction performances of news attributes in a unified way.

7 Conclusions and Future Work

In this paper we propose a unified approach to extract multiple news attributes from news pages by using both the *independent visual features* and the *dependent features*. The extensive experiments show the effectiveness of our approach. We believe it is a promising way to improve the extraction performance by exploiting the layout relation among attributes. In the future, we will try to perform the extraction task on other types of web objects, including web Blog and detailed product object in web pages.

Acknowledgement

We would like to thank the anonymous reviewers for useful comments. This work was supported in part by the China Postdoctoral Science Foundation funded project under grant 20080440256 and 200902014, NSFC (60875033), National High-tech R&D Program (2008AA01Z421) and National Development and Reform Commission High-tech Program of China (2008-2441). Any opinions, findings, conclusions, and/or recommendations in this material, either expressed or implied, are those of the authors and do not necessarily reflect the views of the sponsors listed above.

References

1. Zheng, S., Song, R., Wen, J.-R.: Template-Independent News Extraction Based on Visual Consistency. In: AAAI 2007, pp. 1507–1511 (2007)
2. Reis, D., Golgher, P., Silva, A.: Automatic web news extraction using tree edit distance. In: WWW 2004, pp. 502–511 (2004)
3. Zhai, Y., Liu, B.: Web data extraction based on partial tree alignment. In: WWW 2005, pp. 76–85 (2005)
4. Zhao, H., Meng, W., Wu, Z.: Fully automatic wrapper generation for search engines. In: WWW 2005, pp. 66–75 (2005)
5. Xue, Y., Hu, Y., Xin, G.: Web page title extraction and its application. *Inf. Process. Manage.* 43(5), 1332–1347 (2007)
6. Zhu, J., Nie, Z., Wen, J.-R.: 2D Conditional Random Fields for Web information extraction. In: ICML 2005, pp. 1044–1051 (2005)
7. Zhu, J., Nie, Z., Wen, J.-R.: Simultaneous record detection and attribute labeling in web data extraction. In: KDD 2006, pp. 494–503 (2006)
8. Chang, C.-H., Kayed, M., Girgis, M.R., Shaalan, K.F.: A Survey of Web Information Extraction Systems. *IEEE Trans. Knowl. Data Eng.* 18(10), 1411–1428 (2006)
9. Crescenzi, V., Mecca, G., Meriardo, P.: RoadRunner: Towards Automatic Data Extraction from Large Web Sites. In: VLDB 2001, pp. 109–118 (2001)
10. Liu, B., Grossman, R.L., Zhai, Y.: Mining data records in Web pages. In: KDD 2003, pp. 601–606 (2003)
11. Lafferty, J.D., McCallum, A., Pereira, F.: Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In: ICML 2001, pp. 282–289 (2001)
12. Lu, Y., He, H., Zhao, H., Meng, W., Yu, C.T.: Annotating Structured Data of the Deep Web. In: ICDE 2007, pp. 376–385 (2007)
13. Wang, J., He, X., Wang, C., Pei, J., Bu, J., Chen, C., Guan, Z., Lu, G.: News article extraction with template-independent wrapper. In: WWW 2009, pp. 1085–1086 (2009)
14. Sarawagi, S., Cohen, W.W.: Semi-Markov Conditional Random Fields for Information Extraction. In: NIPS 2004 (2004)
15. Cai, D., Yu, S., Wen, J.-R., Ma, W.-Y.: VIPS: a vision based page segmentation algorithm, Microsoft Technical Report, MSR-TR-2003-79 (2003)
16. Yao, L., Tang, J., Li, J.-Z.: A Unified Approach to Researcher Profiling. In: Web Intelligence 2007, pp. 359–366 (2007)
17. Pinto, D., McCallum, A., Wei, X., Croft, W.B.: Table extraction using conditional random fields. In: SIGIR 2003, pp. 235–242 (2003)
18. Arasu, A., Garcia-Molina, H.: Extracting Structured Data from Web Pages. In: SIGMOD 2003, pp. 337–348 (2003)
19. Zhu, J., Nie, Z., Zhang, B.: Dynamic hierarchical Markov random fields and their application to web data extraction. In: ICML 2007, pp. 1175–1182 (2007)

A Method for Mobile User Profile and Reasoning

Wei Liu and Zhoujun Li

School of Computer Science and Engineering, Beijing University of Aeronautics & Astronautics, Xueyuan Road 37, 100191, Beijing, China
liuwei_xx@sina.com

Abstract. Both of mobile multimedia and mobile Internet are the important development directions of the mobile service. However, it would take great cost by using the high data transmission rate of wireless multimedia communication service. Under the premise of not increasing the investment in hardware, the personalized service could be applied to the mobile service to not only reducing wireless multimedia communication cost but also keeping the quality of mobile service for users. This paper presents a modeling method of mobile phone user profile based on Ontology. This paper founds a model in a method of spatial graph and introduces the theory of interval valued fuzzy sets, brings forward a series of correlative definitions and formulae of founding the model and designs an Algorithm on the Spatial Graph's Establishment and Updating. Then, it also studies the reasoning technology based on the mobile phone user profile and presents a Reasoning Algorithm on the Mobile Phone User Profile. It should be considered that we have made a useful attempt on the study of founding user profile and forecasting users' possible requirements.

Keywords: mobile service, interval-valued fuzzy sets, user profile, spatial graph.

1 Introduction

At present, it would take great cost by using the high data transmission rate of wireless multimedia communication service because the communication service needs large number of scarcity resources of radio spectrum. Now, the personalized service is an important study direction of mobile service. Under the premise of not increasing the investment in hardware, the personalized service could be applied to mobile service to not only reducing wireless multimedia communication cost but also without reducing quality of mobile service for users. It collects users' individuation information in mobile terminals, analyses and forecasts users' requirments, and then, organizes multicast push when the transmission load of Mobile Network is light, and pushes appropriate information resource gained from Internet and lays in these mobile terminals. It not only can greatly reduce the cost of wireless transmission, but also pellucidly offer users with high-speed wireless network resources and personalized service from the user's point of view.

2 Related Works

If we could reasonably forecast users' personalized requirements, it will improve their satisfaction degree and also is the sticking point whether personalized service is applied to mobile service successfully or not. The study of user profile is the base and core of the personalized service[1].

Some researchers have introduced the user model into the mobile communication. Researchers of University Hannover build user profiles for mobile users and capture users' movement position by using cell-ID and Wireless Signal Booster[2]. Giuseppe Araniti integrates user profile with Quality-of-Service(QoS) and studies soft QoS mechanism in wireless multimedia resource[3]. In the 3G network, the researchers of University of California build real-time user group profile and reserve resource for user groups to provide better QoS to different classes of users[4]. Spyros Panagiotakis introduces the information, such as location, into mobile environment to better determine the user's environment[5]. G. Bartolomeo studies how to build user profile for mobile terminal to customize and obtain services safely[6]. Researchers of University of Toronto collect mobile phone user's preference information to provide customized advertisements for them[7]. Under the mobile environment, researchers of Beijing University of Posts and Telecommunications study user profile when the mobile terminal automatically chooses services provided by the mobile provider and the user profile is based on Markov decision process[8]. As a National Key Technological R&D Program, some researchers of Zhejiang University advance a user's SmartShadow model in the pervasive computing environments which adopts Belief-Desire-Intention model to found user profile[9].

These researchers mostly focus on how to acquire environmental information of the users in mobile Internet and customization services. On the base of our previous work[10], we would discuss how to establish the mobile phone user profile and forecast the users' possible requirements.

3 The Requirement Issue

Our system forwardly pushes useful information to the mobile terminal users according to the acquired Internet information. The two sides of the system are the mobile user terminal and the server-side. The work principle of the system is the following: The mobile terminal regularly uploads the user browsing information to the server-side; the server-side mines the user interests and finds the miniature user profile for each user. At the same time, the information acquired from Internet is provisionally stored in the corresponding information repositories, and then, is drawn out in batches according to some strategies. The tree graph model integrates the individual requirements of the mobile terminal users with the recommended information resources on the Internet. Finally, users who have common interest will be organized as a user group model and the system would push corresponding information to the users according to some information such as location. From this system's work principle described above, we can see that

it is the basis and key on judging the system's success or failure of analyzing and accurately predicting the users' real requirements. So it is the important content research of establishing mobile phone user profile and vaguely conferring users' requirements.

We study a model method of mobile phone user profile based on Ontology in this paper. The reasoning is an important constituent part on the study of user profile. We believe that the user profile is only a tool and it needs to combine the reasoning technology to improve its effectiveness. So, only a system combining the both reasonably could predict the requirement of users.

4 User Profile Establishment

The system would build a compact user profile for each user which stores the user's interest profiles. It is not a general description of a user, but a kind of user formal description of algorithm-oriented and with specific data structures. The user profile is composition of spatial graph based on Ontology.

4.1 Expression on Node and Several Formulae

We will present the definitions on space, sub space and node based on Ontology as following.

Definition 1. To suppose G is the topological space, i.e., G is a nonempty set. If some subsets of G are defined as open sets and they meet the following conditions:

- 1) G and ϕ are open sets;
- 2) The union of arbitrary number of open sets is an open set;
- 3) The intersection of limited number of open sets is an open set,

Then, these open sets are called topological structure in G and G is the topological space.

Definition 2. If $G' \subseteq G$, then, G' is the topological subspace of G .

Definition 3. A four-tuples is used to express the node based on Ontology, ON (Md, R, AT, IS). The meaning of every tuple is listed as follows.

Md: The meta information description on the node.

R: Relation. It is a two-tuple, $\langle R_r, R_w \rangle$. R_r is the relations on Ontology which includes hypnym, hyponym, synonymy and antonym, etc. R_w is the words that relative to the relations in the supported Ontology libraries. The words can be extended.

AT: The attribute of the node.

IS: The instance of the node.

We should consider the establishing and updating of the nodes and edges of the spatial graph based on Ontology. Firstly, the system would cluster the user data uploaded from the mobile terminals. Then, the clustering result center would be

compared with nodes of the spatial graph and do a mapping to the closed interval, [0,1] based on the interval-valued fuzzy sets. Because we seldom acquire an accurate value from clustering and the clustering result center can be expressed in an interval-valued fuzzy set, $A(x) = [A^-(x), A^+(x)]$, $x \in G$. In the same way, the node in this paper can also be expressed with $B(x_i) = [B^-(x_i), B^+(x_i)]$, $x_i \in G$. The i is a certain node in the spatial graph.

Researchers point out that it is also an interval value of the corresponding topology relation degree in the fuzzy domain if the fuzzy domain is expressed in two interval-valued fuzzy sets[11]. So the problem of comparing clustering result center with nodes can be solved in the intersection degree between the two fuzzy domains. It is showed as follows of the Formula on Interval-valued Fuzzy Set Mapping between Clustering Center and Nodes:

$$P_{ci} = [\bigvee_{x \in G} \{A^-(X) \wedge B_i^-(X)\}, \bigvee_{x \in G} \{A^+(X) \wedge B_i^+(X)\}] \quad (1)$$

P_{ci} expresses the comparative value of clustering result center and nodes. $P_{ci} \in [I]$. The $[I]$ refers to the unit closed interval. If $P_{ci} = [1,1]$, then, the two fuzzy domains must intersect. If $P_{ci} = [0,0]$, then, the two fuzzy domains must disjoint absolutely.

Here, if the P_{ci} is bigger than a certain threshold, then, the information represented by the clustering result center would belong to the node and be added to it. If the P_{ci} exceeding the threshold of more than one node, then, it would be added to all of these nodes and the values of them would be amended, too. If not exceeding any threshold, then, the clustering center would be added to the spatial graph as a new node. Then, it should judge whether mount a new edge or not between the new node and each old node. The following is the Formula on New Edge Setting Judging:

$$S_i = \begin{cases} \frac{2}{\pi} \cdot \arctan \frac{P_i^+ + P_i^-}{2(P_i^+ - P_i^-)}, & P_i^+ \neq P_i^-, S_i \in I \\ P_i^+ = P_i^-, & P_i^+ = P_i^- \end{cases} \quad (2)$$

To suppose $P_{ci} = [p_i^-, p_i^+]$. The S_i should be related to the two factors: $0 < \frac{P_i^+ + P_i^-}{2} < 1$ and $0 < P_i^+ - P_i^- < 1$.

Then, it would mount a weight value on the new edge. To judge whether mount a new edge or not based on S_i gained from Formula(2) and to endow the new edge mounted with a weight value:

$$f_i = \mu * S_i - \ell / W_i, f_i \in (0,1) \quad (3)$$

The f_i relates to the two factors, S_i and W_i . There is f_i , S_i and W_i changing with the same tendency. The ℓ is a coefficient, $0 < \ell < W_i < 1$, $\ell \in R$, we will carefully choose ℓ to keep ℓ / W_i in (0,1). The μ is a coefficient, too. $\mu > 0$, $\mu \in R$. We will also carefully choose μ and ℓ to keep f_i in (0,1).

Then, we would consider to updating the weight value of the old edges. If more than one old node is amended, then, the nodes would be checked whether having

edges among them or not. If it is, the weights of the old edges would be modified, if not, a new edge should be created. If only an old node is amended, then, only its edges would be amended. If it exceeds a certain threshold, a new edge would be created, and if it is less than another threshold, then, the edge would be deleted. At the same time, the time counter of the edge would be amended, too. Thus, it is accomplished of expressing and updating on the spatial graph.

The Formula on Time Decay of Edge is considered related to the two factors. One is the time from the edge is established to now. The other is the prompting to the edge in this time, that is, it plays an enhanced role of using the nodes every time. The Formula on Time Decay of Edge is showed as following:

$$W = \begin{cases} e^{-\frac{\ln(t+1)}{k \times S}}, & t \neq 0 \\ 1, & t = 0 \end{cases} \quad (4)$$

We suppose t expresses the time from the edge is founded to now and the value of W should reduce along with t increasing. To suppose that the s expresses the prompting, $s \in N$, and the value of W should increase as soon as being stimulated. To suppose the W is the weight on time decaying of edge and it is a function on t and s , $W \in (0,1]$. The k is a coefficient, $k > 1$.

4.2 Algorithm on the Spatial Graph's Establishment and Updating

Algorithm 1. Algorithm on the Spatial Graph's Establishment and Updating

Input: D_t : the useful data of the acquired user's behaviors

G_r : a spatial graph

Output: G_r' : the amended spatial graph

1: According to a fuzzy clustering algorithm, to fuzzy cluster D_t to generate a clustering result center, D_c .

2: **for all** $i : i \in [1..n]$ **do** // There are n nodes in G_r which would be compared with D_c one by one.

3: To compare D_c with G_{ri} , then, according to the formula(1), the comparing result would be mapped to $[I]$ based on the theory of interval-valued fuzzy set // G_{ri} expresses a certain node.

//To compare P_{ci} acquired from the formula(1) with the threshold, α , to judge whether creating a new node or amending the old nodes. It is showed as follows.

4: **if** $P_{ci} \geq \alpha$ **then** // If it exceeds the threshold, α , the old nodes would be amended.

5: **if** only one node has $P_{ci} \geq \alpha$ **then**

6: To add D_c to G_{ri} and amend the value of G_{ri} //If exceeding the threshold, α , D_c can be considered belonging to G_{ri} and would be added to the node.

7: *OldSideOne*

8: **end if**

9: **if** more than one node have $P_{ci} \geq \alpha$ **then**

10: To suppose $\exists i_1, i_2, \dots$, and $i_1 \neq i_2 \neq \dots$, which have the corresponding nodes, G_{ri1}, G_{ri2}, \dots

```

11:      To add  $D_c$  into  $G_{ri1}, G_{ri2}, \dots$  one by one and amend the values of
 $G_{ri1}, G_{ri2}, \dots$ 
12:      OldSideNotOne
13:      end if
14:  else //If it is less than the threshold,  $\alpha$ , a new node would be generated.
15:      To add  $D_c$  into the spatial graph,  $G_r$ , as a new node
16:      To generate the node based on the data,  $D_t$ , and the node definition
17:      NewSide
18:  end if
19: end for
20: NewSide
21:  for all  $i : i \in [1..n]$  do //To judge whether founding new edges among
a new node and each old node or not based on the formula(2). It is showed as
follows.
22:      To compare it with an appointed threshold in advance,  $\beta$ , based on
the formula(2).
23:      if  $S_i \geq \beta$  then
24:          It mounts a new edge between the new node and the old node.
25:          To calculate the weight value of the new edge and assign it to the
new edge based on the formula(3).
26:          It mounts the direction of the edge according to the relation between
the two nodes.
27:          end if
28:      end for
29: OldSideNotOne //To amend and update the old edges between these nodes
when there are more than one old node being amended.
30:  To check whether any edge exists among the amended old nodes,  $G_{ri1}, G_{ri2},$ 
 $\dots$ , or not.
31:  if any edge existing then
32:      To amend the weight of the old edge based on the formula(4),  $s:=s +$ 
1. Then, to calculate the amended weight value of the edge according to the
formula(3).
33:  else
34:      To establish a new edge.
35:      To calculate  $P_{ci}$  over again according to the amended old node by using
the formula(1). Then, it calculates the weight of the new edge by using formulae
(2), (3) and (4).
36:  end if
37: OldSideOne //To amend and update the old edge of the node when there
is only one old node,  $G_{ri}$ , being amended in  $G_r$ .
38:  It amends the edges which the amended node  $G_{ri}$  has owned.
39:  To use the formula(4),  $s:=s + 1$ , then, use the formula(3) to get a weight
value of the amended edge.
40: Return  $G_r'$ 

```

4.3 Algorithm Analysis

There are five important characters on an algorithm, that is, input data, output data, determinacy, finiteness and effectiveness. An algorithm should be feasible which could meet the five characters given above. There is the input data, which are D_t , the gained useful data on the user's behaviors, and G_r , a spatial graph, in the Algorithm 1 presented in the paper. There is also the output data, that is, G_r' , the amended spatial graph. The effectiveness of an algorithm couldn't be proved with a better means in theory at present[13]. The meaning of algorithm effectiveness is that each step of an algorithm should be executed effectively, that is, operations described in an algorithm could realize by executing limited steps of actualized basic operations. If the algorithm analysis on determinacy and complexity is based on each sentence and elementary operation, then, it would indirectly prove its effectiveness. So the following will respectively analyze the algorithm 1 in the two aspects: determinacy and finiteness that is mainly time complexity.

Analysis on Algorithm Determinacy

The determinacy is that each step of an algorithm should be certain. The algorithm would be determinate if it meets the well-ordered principle[12], [13].

Theorem 1. If a clause set G of a well-ordered can infer $X_1 \prec X_n$, that is, $X_1 \rightarrow X_n$, then, the deduction process could be represented as $G \cup \{X_1, \sim X_n\}$, an insatiable clause set.

Demonstration: To see the demonstration on theorem 3.35 in reference [13].

Theorem 2. To suppose P is the beginning sentence of an algorithm and Q is its end statement. If an algorithm is certain, then, the $P \rightarrow Q$ can be inferred from the clause set G .

Demonstration: To see the demonstration on theorem 4.2.20 in reference [14].

Deduction 1. If an algorithm is certain, then, it could be represented as $G \cup \{P, \sim Q\}$, an insatiable clause set.

The following is to construct a clause set G of this algorithm and analyze its sentences:

1) The sentence 1 is the beginning of the algorithm which is expressed with P ;

2) It is a loop structure of the sentences 2-19. Hereinto, the sentence 3 is expressed with A_1 for it is an in-order execution relation between it and the following sentences. The sentences 4-18 is a nested branching optional structure which embeds *IF* only one node has $P_{ci} \geq \alpha$ (the sentences 5-8), the sentence 6 and 7 are respectively expressed as A_2 and A_3 for they are an in-order execution relation; and *IF* more than one node have $P_{ci} \geq \alpha$ (the sentences 9-13), the sentence 10, 11 and 12 are respectively expressed as A_4 , A_5 and A_6 for they are also an in-order execution relation; *ELSE* $P_{ci} < \alpha$ (the sentences 14-18), and the sentences 15, 16 and 17 are respectively expressed as A_7 , A_8 and A_9 for they are also an in-order execution relation;

3) The sentences 20-28 is the invoked *NewSide*: The sentence 20 is an entry function and is expressed with A_{10} . The sentences 21-28 is a loop structure. Hereinto, the sentence 22 is expressed with A_{11} for it is an in-order execution relation between it and the following sentences and the sentences 23-27 is an *IF* sentence which is expressed with A_{12} ;

4) The sentences 29-36 is the invoked *OldSideNotOne*: The sentence 29 is an entry function and is expressed with A_{13} . The sentence 30 is expressed with A_{14} for it is an in-order execution relation between it and the following sentences. The sentences 31-36 is a branching optional structure, *IF* (the sentences 31-32) is expressed with A_{15} , *ELSE* (the sentences 33-36) is expressed with A_{16} ;

5) The sentences 37-39 is the invoked *OldSideOne*: The sentence 37 is an entry function and is expressed with A_{17} . The sentences 38 and 39 are respectively expressed as A_{18} and A_{19} for they are an in-order execution relation;

6) The sentence 40 is the end of the algorithm which is expressed with Q .

From the analysis on the algorithm sentences above, we can prove its determinacy.

Demonstration: The clause set of the algorithm 1

$$\begin{aligned}
 G = & \{ (P \rightarrow A_1), \bigvee_{i=2,4,7} (A_1 \rightarrow A_i), A_2 \rightarrow A_3, (A_4 \rightarrow A_5) \wedge (A_5 \rightarrow A_6), (A_7 \rightarrow \\
 & A_8) \wedge (A_8 \rightarrow A_9), A_3 \rightarrow A_{17}, A_6 \rightarrow A_{13}, A_9 \rightarrow A_{10}, (A_{10} \rightarrow A_{11}) \wedge (A_{11} \rightarrow \\
 & A_{12}), A_{13} \rightarrow A_{14}, \bigvee_{i=15,16} (A_{14} \rightarrow A_i), (A_{17} \rightarrow A_{18}) \wedge (A_{18} \rightarrow A_{19}), \\
 & \bigvee_{i=12,15,16,19} (A_i \rightarrow Q) \} \\
 = & \{ \sim P \vee A_1, \sim A_1 \vee A_2, \sim A_1 \vee A_4, \sim A_1 \vee A_7, \sim A_2 \vee A_3, \sim A_4 \vee A_5, \\
 & \sim A_5 \vee A_6, \sim A_7 \vee A_8, \sim A_8 \vee A_9, \sim A_3 \vee A_{17}, \sim A_6 \vee A_{13}, \sim A_9 \vee A_{10}, \\
 & \sim A_{10} \vee A_{11}, \sim A_{11} \vee A_{12}, \sim A_{13} \vee A_{14}, \sim A_{14} \vee A_{15}, \sim A_{14} \vee A_{16}, \sim A_{17} \\
 & \vee A_{18}, \sim A_{18} \vee A_{19}, \sim A_{12} \vee Q, \sim A_{15} \vee Q, \sim A_{16} \vee Q, \sim A_{19} \vee Q \}
 \end{aligned}$$

Then, it can be known that there is $P \rightarrow Q$ in the clause set G according to the well-ordered definition and the Theorem 1, that is, the deduction process could be represented as $G \cup \{P, \sim Q\}$, an insatiable clause set. In line with the Theorem 2 and Deduction 1 again, it is proved of its determinacy.

Analysis on Algorithm Time Complexitys

To suppose the $L_0 = \max(|D_j|), j = 1, 2, \dots, r$, the D_j expresses once execution time of the process that it fuzzy clusters D_i to generate a clustering result center, D_c , according to a fuzzy clustering algorithm. To suppose the L_1 is the time on amending an old node and the L_2 is the time on founding a new node. Then, to suppose the L_3 expresses the time on mounting a new edge for a new node which includes mounting an edge and gaining its weight and direction. To suppose the L_4 is the time on amending and updating the old edges which have been owned by the amended old nodes and the L_5 expresses the time on mounting a new edge between two amended old nodes.

To analyze this algorithm, we can find that there are two steps in its implementation process: the first is the pretreatment on the spatial graphics which is D_j and the second is the treatment on the spatial graph which includes dealing with both nodes and edges, that is, L_1, L_2, L_3, L_4 and L_5 . In connection with

a certain D_j , all of the above five implementations wouldn't appear at the same time, while there may be two or three. Hence, it is necessarily greater than the actual algorithm complexity of the calculated algorithm complexity according to the five cases happening at the same time.

① The instance on only one old node in the G_r being amended.

There are n nodes in the G_r . This instance includes the max time about the pretreatment on the spatial graph, selecting and amending an old node from the G_r and amending all of the existent old edges between this old node and all of the other old nodes in the G_r , that is, $L_0 + L_1 + (n - 1)L_4$;

② The instance on not only one old node in the G_r being amended.

This instance includes the max time about the pretreatment on the spatial graph, selecting and amending more than one old node from the G_r , amending all of the existent old edges and mounting new edges between every amended old node and all of the other old nodes in the G_r . We suppose that an amended old node would both be amended old edges and mounted new edges once each with all of the other old nodes. In fact, both of them are relatively prime, so we can take the average of both, that is, $L_0 + n[L_1 + \frac{n-1}{2}(L_4 + L_5)]$;

③ The instance on founding a new node in the G_r .

This instance includes the max time about the pretreatment on the spatial graph, founding a new node and potential mounting new edges between this new node and all of the old nodes in the G_r , that is, $L_0 + L_2 + nL_3$;

④ The time on completing basic operations of the arithmetic in a whole execution, namely, in a process of founding and updating the user profile, is less than

$$L_0 + L_1 + (n - 1)L_4 + L_0 + n[L_1 + \frac{n-1}{2}(L_4 + L_5)] + L_0 + L_2 + nL_3 \\ = 3L_0 + (n + 1)L_1 + L_2 + nL_3 + \frac{1}{2}(n^2 + n - 2)L_4 + \frac{1}{2}(n^2 - n)L_5$$

Over here, the $L_i (i = 1, 2, 3, 4, 5)$ is the time of accomplishing a certain basic operation which can be regarded as a constant. The $L_0 = \max(|D_j|), j = 1, 2, \dots, r$, is the upper limit of pretreatment on the spatial graph and that it is only a minimum probability event in practice of the D_j could gain the value of L_0 , so, it can be regarded as a constant, too. Therefore, the equation above is the same order with n^2 . If it is noted down $T(n)$, then, there is $T(n) = O(n^2)$. Because the $T(n)$ is gained when $L_i (i = 1, 2, 3, 4, 5)$ appears at the same time which is surely greater than the actual situation, the time complexity of the algorithm could be expressed as $T(n) = o(n^2)$.

The above analyzes and proves the Algorithm 1 from the aspect of theory. Hence, it is easy to see that the algorithm is proved of the traits of determinacy, effectiveness and low time complexity.

5 User's Requirements Reasoning

After establishing a user profile, the system needs analyze and forecast the user's actual possible requirements and termly upload to the system based on the nodes which have changed in a cycle. The reasoning background is acquired as the analysis and summary on the user's online behaviors which is stored in the

user profile. That is to say, the concrete manifestation of the background is just the spatial graph based on Ontology which is the main basis when the system reasoning the user's requirements.

5.1 Reasoning Mechanism

To suppose the changed nodes in a spatial graph is Q , then, we reason the Q to acquire possible results according to a user's actual requirements which is expressed as Q' . The f expresses the reasoning process, the Q is regarded as the reasoning antecedent and premise, and the Q' is regarded as the reasoning consequent and conclusion. We suppose \widehat{A}_i^l as a link of the reasoning process, that is, a node of a certain selected reasoning path. Here, the $l \in [1, 2, \dots, k]$ expresses the count of a certain reasoning path, the k expresses the total number of all possible paths in the reasoning process, the $i \in [1, 2, \dots, n]$ expresses the count of a certain node in a certain reasoning path and the n expresses the number of nodes in this path. Then, the reasoning process can be expressed as follows:

$$\begin{array}{l}
 f : Q \implies Q' \\
 Q \mapsto \bigvee_{l=1}^k \left(\prod_{i=1}^n \widehat{A}_i^l \right) \\
 \text{-----} \\
 Q'
 \end{array} \tag{5}$$

Among them, the \mapsto expresses the detrusion symbol, the \bigvee expresses the select symbol which refers to choose a certain reasoning path and the \prod expresses the orderly path of the nodes in a certain reasoning path. It can be seen that the reasoned Q' maybe is not uniqueness relative to a certain Q which is related to the choice of $\prod_{i=1}^n \widehat{A}_i^l$.

5.2 Reasoning Algorithm on the Mobile Phone User Profile

Algorithm 2. Reasoning Algorithm on the Mobile Phone User Profile

Input: G_r : a spatial graph

Output: I_i : the information intersection of all nodes in a certain selected path

- 1: **for all** $i : i \in [1..n]$ **do** //There are i number of changed nodes in G_r .
- 2: **if** N_i is a new node **then**
- 3: *FindRoute*
- 4: **else**
- 5: **if** $P_{ci} \geq \lambda$ **then** //If the value, P_{ci} , of the node N_i is bigger than a certain threshold value, λ , appointed in advance, then the N_i is just the destination nodes.
- 6: *FindRoute*
- 7: **end if**
- 8: **end if**

```

9: end for
10: FindRoute //To find the optimal path as following.
11: Starting from the node  $N_i$  to find out all of its upper nodes  $N_j^{(l)}$ ,  $j \in [1..m]$ ,  $l \in [1..t]$  and the inter-connected directed edges among these nodes  $S_k^{(l)}$ ,  $k \in [1..r]$ .
    //The  $N_i$  has  $m$  number of upper nodes. There are  $r$  number of directed edges connecting between  $N_i$  and these upper nodes. At the same time, the  $l$  plays part in layering on these upper nodes and directed edges in the light of different distances from these nodes to  $N_i$  and there are  $t$  layers.
12: begin
13:   for all  $l : l \in [1..t]$  do
14:     To find out  $\max(S_k^{(l)})$  // To find out the node with the biggest value among the directed edges' weight in a same layer.
15:     To note down the corresponding node  $N_j^{(l)'}$  // To note down the corresponding upper node of the edge.
16:     To continue searching the next from the upper nodes of  $N_j^{(l)'}$ 
17:   end for
18:   To record  $R_t = \{N_j^{(l)'}\}$  and obtain the optimal path.
19:   To draw out the user's information,  $N_j^{(l)''}$ , contained in  $N_j^{(l)'}$ , the nodes of the path.
20:    $I_i = \cap N_j^{(l)''}$ ,  $l \in [1..t]$  //To draw out the user's information contained in the nodes of the path and acquire the intersection.
21: end
22: Return

```

5.3 Algorithm Analysis

In the Algorithm 2, there is the input data, which is G_r , a spatial graph. There is also the output data, that is, I_i , the information intersection of all nodes in the selected path.

Analysis on Algorithm Determinacy

The determinacy proof of the Algorithm 2 is as same as the Algorithm 1 and we could not prove it in detail as space is limited.

Analysis on Algorithm Time Complexity

For the same reason, we couldn't analyze the time complexity of the algorithm 2, either, which is a polynomial expression $T(n) = O(n(L_0 + L_1 + L_2 + L_3))$.

6 Conclusion

The paper focuses on how to found mobile phone user profile and forecast users' possible requirements. In this paper, we present a model method of mobile phone user profile based on Ontology and introduce the theory of interval valued fuzzy sets. The proposed method brings forward a series of correlative definitions and formulae on founding the model and designs an Algorithm on the Spatial Graph's

Establishment and Updating. Then, we also study the reasoning technology based on the mobile phone user profile and present a Reasoning Algorithm on the Mobile Phone User Profile. For the future, we should make further study in depth on the aspects such as dynamic user group model and prediction accuracy measurement on user group model.

References

1. Cufoglu, A., Lohi, M., Madani, K.: A Comparative Study of Selected Classification Accuracy in User Profiling. In: Seventh International Conference on Machine Learning and Applications, San Diego, CA, pp. 787–791 (2008)
2. Erbas, F., Kyamakya, K., Steuer, J., Jobmann, K.: On the User Profiles and the Prediction of User Movements in Wireless Networks. In: The 13th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, pp. 2282–2286 (2002)
3. Araniti, G., De Meo, P., Iera, A., Ursino, D.: Adaptively Controlling the QoS of Multimedia Wireless Applications Through “User Profiling” Techniques. *IEEE Journal on Selected Areas in Communications* 21, 1546–1556 (2003)
4. Pandey, V., Ghosal, D., Mukherjee, B.: Exploiting User Profiles to Support Differentiated Services in Next-Generation Wireless Networks. *IEEE Network*, 40–48 (2004)
5. Panagiotakis, S., Koutsopoulou, M., Alonistioti, A., Thomopoulos, S.: Context Sensitive User Profiling for Customised Service Provision in Mobile Environments. In: IEEE 16th International Symposium on Personal, Indoor and Mobile Radio Communications, pp. 2014–2018 (2005)
6. Bartolomeo, G., Berger, F., Eikerling, H.J., Martire, F., Salsano, S.: Handling User Profiles for the Secure and Convenient Configuration and Management of Mobile Terminals and Services. In: Proceedings of the 16th International Workshop on Database and Expert Systems Applications, pp. 272–277 (2006)
7. Bila, N., Cao, J., Dinoff, R., Ho, T.K., Hull, R., Kumar, B., Santos, P.: Mobile User Profile Acquisition through Network Observables and Explicit User Queries. In: The Ninth International Conference on Mobile Data Management, Beijing, China, pp. 98–107 (2008)
8. Huang, H., Wang, Y.: Modeling of User Preference Based on Agent. *Journal of Harbin Institute of Technology* 39, 1163–1165 (2007)
9. Zhang, L., Pan, G., Li, S., Wu, Z.: SmartShadow: A New Model of Pervasive Computing. In: The 4th Joint Conference on Harmonious Human Machine Environment, Wuhan, China, pp. 175–182 (2008)
10. Chen, J., Liu, W.: A Modeling Method of User Profile Based on Ontology. *Journal of Computer Research and Development* 44, 1151–1159 (2007)
11. Yu, Q., Liu, D., Ouyang, J.: Topological Relations Model of Fuzzy Regions Based on Interval Valued Fuzzy Sets. *Acta Electronica Sinica* 33, 187–189 (2005)
12. Zhou, P.: Design and Analysis of Computer Algorithm. Machine Industry Publishing Company, Beijing (1985)
13. Du, Y.: Study and Implementation on Intelligent Action of Search Engine. Southwest Jiaotong University, Chengdu (2005)
14. Qin, X.: Studies on Intelligent Office Information System. Southwest Jiaotong University, Chengdu (2003)

Evaluating Importance of Websites on News Topics

Yajie Miao, Chunping Li, Liu Yang, Lili Zhao, and Ming Gu

School of Software, Tsinghua University
Beijing 100084, China

yajiemiao@gmail.com, cli@tsinghua.edu.cn,
bluewillowwind@gmail.com, zhaoll07@mails.tsinghua.edu.cn,
guming@tsinghua.edu.cn

Abstract. In this paper, we study a novel problem which we refer to as *News Website Evaluation* (NWE). Given a collection of news articles, NWE is primarily concerned with evaluating the importance of their websites with respect to specific news topics. This general problem subsumes many interesting applications including news tracking and website ranking. To solve this problem, we first propose a *Topic-oriented Website Evaluation Model* (TWEM) which exploits various forms of information and combines them in a unified computation framework. Then, considering the special characteristics of news articles, we incorporate an *article merging* operation into TWEM and present the *merge-TWEM* model. The experimental results show that the proposed models perform significantly better than competitive baseline systems, and can serve as effective solutions to the News Website Evaluation problem.

Keywords: News Website Evaluation, Website Ranking, News Articles, Web Mining.

1 Introduction

As online news pages are accumulating to an intractably huge size, how to retrieve desired information has become an increasingly important issue. Under such circumstances, news search is attracting intensive attention from research community and commercial organizations. Some web services such as Google News have been able to provide users satisfactory ranking results of news pages. However, in some cases, we are also interested in the ranking of websites on specific news topics. For instance, we have no idea whether CNN is more authoritative than CBS News in reporting “Copenhagen Conference”, although from Google News we can find the important news pages on this topic.

Page ranking has been a traditional focus of information retrieval, and a number of methods have been proposed for this task. However, there are yet no mechanisms with which we can rank websites according to specific news topics. In this paper, we define and study a novel problem which is referred to as News Website Evaluation (NWE). Given a collection of news articles¹, the task of NWE is to evaluate the

¹ News article and news page are equivalent concepts in this study.

importance of the websites that these articles belong to. This problem potentially has many application scenarios. A typical example involves tracking analysis of news reports on the web. For recognizing the propagation pattern of news, we are more interested in the spreading process among different websites rather than pages. In this situation, a key factor which has to be considered is the relative importance of websites on this news topic.

As for solutions to this problem, we first propose a Topic-oriented Website Evaluation Model (TWEM). To achieve desirable performance, TWEM takes advantage of various forms of information. Specifically, TWEM considers interdependency between websites and news articles, as well as mutual *support* among news articles. In addition, the inherent popularity of websites is also considered when we infer their final importance scores. Then, we adapt the TWEM model to the special features of news articles by introducing an article merging operation. Article merging aims to merge similar news articles into *super-articles*. We propose another model, named merge-TWEM, to combine TWEM and article merging together. We conduct extensive experimental studies to test the proposed models. Experimental results on the real dataset show that both TWEM and merge-TWEM outperform the baseline systems to a great extent. Moreover, performance comparison reveals that merge-TWEM achieves better results than TWEM, and thus demonstrates that the article merging operation indeed takes effects in boosting the performance of TWEM.

The rest of the paper is organized as follows. Section 2 reviews previous work which is related with this study. Section 3 presents the proposed models. In section 4, we give and discuss the experimental results. We have the conclusion and future work in Section 5.

2 Related Work

2.1 Page Ranking

Page ranking is a well studied problem in information retrieval and web mining. PageRank [7] and HITS[1] are two well-known models for ranking web pages based on link analysis. Besides link structures, topical information has been exploited for designing more sophisticated ranking models. Haveliwala et al. [8] proposed the Topic-sensitive PageRank model to combine topical analysis with PageRank. In this model, some topics are selected from predefined categories and a biased PageRank vector is computed for each topic. The final score for each page is got by summing elements of all the PageRank vectors pertaining to different topics. Chakrabarti et al. [10] exploited the anchor texts of hyperlinks to assign each hyperlink a topical weight. This topical weight is employed in the computation process of HITS. Bharat et al. [9] gave each node in HITS a relevance weight which is defined as the similarity of this node's document to the topic query. This relevance weight is used to regulate hub and authority scores computed by HITS. Nie et al. [11] proposed the Topical HITS and Topical PageRank models in which topical information is incorporated into HITS and PageRank in a probabilistic way. For each page, they calculated a score vector to distinguish the contributions from different topics. Their models outperform other approaches and we keep the characteristics of the basic HITS and PageRank unchanged.

Other forms of information are also explored for web ranking. Wang et al. [12] proposed to use media focus and user attention information to rank news topics within a certain news story. Fernandes et al. [13] proposed to use block information in web pages to get better ranking results. Dou et al. [14] introduced methods which incorporate anchor texts into web search and achieve better retrieval performance. Guo et al. [15] built a Bayesian based click chain model (CCM) for mining web search click logs, which is helpful for improving the results of web search.

2.2 Evaluation of Websites

Another line of related work focuses on conducting evaluations on websites. Inspired by the idea of HITS, Yin et al. [2, 3] proposed the TRUTHFINDER model for identifying trustworthy websites and correct facts on the web. Based on the interdependency between websites and facts, an iterative computation method is used to calculate the trustworthiness of websites and correctness of facts. Dai et al. [6] proposed a trust model to determine the trustworthiness of data providers (websites). They made use of various features, such as data similarity, data conflict, path similarity and data deduction, for calculating the trust scores of websites. Although these models are proved to be effective in deciding whether a website is trustworthy on a subject, they are unable to provide information about whether a website is important on a specific topic, e.g., a news story.

Liu et al. [4] proposed a BrowseRank model to exploit user browsing behavior data for ranking web pages. When ignoring the transitions between pages in the same website, BrowseRank can give the ranking results for websites. Zhu et al. [5] introduced the ClickRank model for estimating web page and website importance from browsing information. In their model, the score for a website is the sum of ClickRank values of its web pages. Gao et al. [16] designed a model to compute the weights of websites and web pages at the same time. However, they ignored critical factors such as website popularity and web page merging.

3 The Proposed Models

In this section, we present two models, i.e., TWEM and merge-TWEM, for the NWE problem. Before going to the details, we first give some formal definitions. For a news topic t , we denote the set of news articles as $A = \{a_i\}$ whose websites comprise the website set $W = \{w_i\}$. Since several articles can belong to the same website, we have $|A| \geq |W|$.

We define the **topical importance** of w_i , denoted as $imp(w_i)$, is a value between 0 and 1 which indicates its relative importance on the topic t . The higher this value, the more important w_i . Then the task of the NWE problem can be further formulated as inferring topical importance of each website in W , and ranking these websites according to their topical importance values.

3.1 TWEM

The TWEM model mainly consists of two steps: *dynamic computation* and *popularity incorporation*. In this subsection, we describe them in detail.

Dynamic computation. For the news topic t , there are usually a large number of articles related with it. It is clear that several news articles can belong to the same website. Also, we assume in this study that one single article can also belong to multiple websites. In order to verify this point, we give an example in Figure 1. The news article a_1 is on website w_1 , and the article a_2 (on another website) has issued important information on this topic. Besides its own contents, a_1 probably gives a hyperlink to a_2 , which is a common case in news articles and websites. In this situation, the article a_2 can be viewed to be contained in w_1 as well, in the sense that a_2 can be accessed from w_1 . Based on this assumption, we can conclude that there is actually a “many-to-many” mapping between websites and articles. An interdependency relationship exists between websites and news articles: an article (like a_2) is considered to be important if it belongs to many important websites; a website (like w_1) is considered to be important if it contains many important articles. Then, an iterative computation method like HITS [1] can be used to calculate the scores of the websites. The websites and articles are organized into a bipartite graph, where the hub nodes are websites and authority nodes are news articles. There is a link from website w_1 to article a_1 if a_1 belongs to w_1 .

Besides relations with websites, news articles themselves have influence on each other. Intuitively, if the contents of one article are similar with many others, this article can be considered to be *supported* by others, and thus should be assigned a higher importance score. The support from article a_i to article a_j , denoted as $\text{sup}(i, j)$, is defined as the amount of importance that should be added to a_j if we know a_i is important. Then, the score of an article comes from two sources: websites as hubs and support from other articles. This idea is captured by the model presented in Figure 2. The authority scores for the articles and the hub scores for the websites are calculated iteratively as follows.

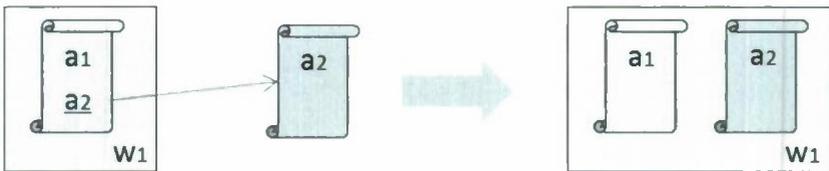


Fig. 1. An example of the “many-to-many” relationship between websites and news articles

$$Auth^{n+1}(a_i) = \frac{\sum_{j=1}^r Hub^n(w_j)}{r} + \sum_{a_k \in A} sup(k,i) \cdot Auth^n(a_k)$$

$$Hub^{n+1}(w_i) = Hub^n(w_i) + \frac{\sum_{j=1}^s Auth^n(a_j)}{s}, \tag{1}$$

where $Auth^n(a_i)$ is the authority score for article a_i in n^{th} iteration, $Hub^n(w_i)$ is the hub score for website w_i in n^{th} iteration, r is the number of websites containing a_i , and s is the number of articles belonging to w_i . $sup(k,i)$ represents the support from a_k to a_i , and is defined in the follow formula.

$$sup(k,i) = \frac{sim(a_k, a_i)}{\sum_{a_j \in A} sim(a_k, a_j)}, \tag{2}$$

where $sim(a_k, a_i)$ is the text similarity between article a_k and article a_i . From a stochastic perspective, $sup(k,i)$ is actually the probability of transiting from a_k to a_i on the graph. To avoid self-transition, we define $sup(i,i) = 0$. After each iteration, we transform hub and authority weights using the function $f(x) = 1 - \exp(-x)$ in order to smooth the values, and to keep the weights between 0 and 1.

From the above computation, we can see that the authority or hub scores are updated many times until the iterative process converges. This is why we call this step dynamic computation. The convergence is achieved when the difference between the scores computed at two successive iterations falls below a given threshold.

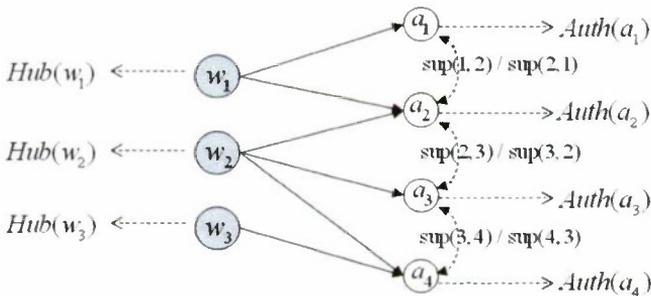


Fig. 2. The TWEM model

Popularity incorporation. In addition to the hub scores derived from their articles, websites themselves have popularity which is independent of specific topics. Popularity is the extent to which websites are popular among the public, e.g., Yahoo is a much popular portal website. Intuitively, if a website is quite popular, users are more likely to issue important information on it in order to attract the attention of others. The accumulation of important articles in turn makes this website important on certain news topics. Therefore, the popularity of websites also has an influence on their

topical importance. A possible way to measure popularity quantitatively is to utilize the website ranks from Alexa². Alexa provides detailed ranks of websites based on page view and traffic. If the Alexa rank of the website w_i is $aRank(w_i)$, its popularity is defined as

$$Pop(w_i) = 1.0 - \frac{aRank(w_i)}{\max\{aRank(w_1), aRank(w_2), \dots, aRank(w_N)\}} \tag{3}$$

where N is the total number of websites in the website collection W .

Final topical importance. we combine the hub score and popularity together, and obtain the topical importance for website w_i , that is,

$$imp(w_i) = \alpha \cdot Pop(w_i) + (1 - \alpha) \cdot Hub(w_i), \tag{4}$$

where α is a factor to control the balance between the popularity and the hub score.

3.2 merge-TWEM

News articles commonly cite contents from each other, especially from authoritative sources, e.g., some news agencies. An example is presented in Figure 3 where the two news articles have identical contents which are originally released by the Associated Press. This citation makes news articles on the same topic usually show great similarity. Because of the support among articles in dynamic computation, a group of very similar articles will prompt each other and obtain unfairly high scores.

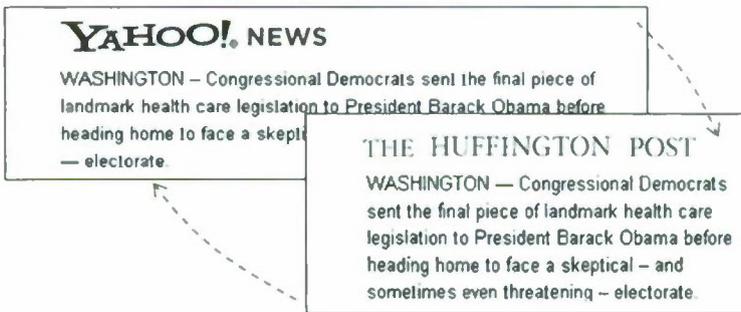


Fig. 3. An example of citation among news articles

To address this problem, we propose a merge-TWEM model which extends TWEM to include an article merging operation. In particular, if the similarity between two articles exceeds a predefined threshold, we merge them into a single *super-article*. A super-article is an article group in which every two *members* have similarity over the threshold. After this merging is conducted, we get the set of super-articles $SA = \{sa_i\}$, where each sa_i is associated with a group of members (articles), i.e.,

² www.alexa.com

$$sa_i = \{a_{i1}, a_{i2}, \dots, a_{i|sa_i|}\}, \tag{5}$$

where a_{ij} is the j^{th} member in super-article sa_i , $|sa_i|$ is the total number of members contained by sa_i . A super-article belongs to all the websites of its members. Figure 4 gives a description of the merge-TWEM model.

Then the iterative computation process for authority scores of super-articles and hub scores of websites can be formulated as follows.

$$Auth^{n+1}(sa_i) = \frac{\sum_{j=1}^z Hub^n(w_j)}{z} + \sum_{sa_k \in SA} msup(k, i) \cdot Auth^n(sa_k)$$

$$Hub^{n+1}(w_i) = Hub^n(w_i) + \frac{\sum_{j=1}^v Auth^n(sa_j)}{v}, \tag{6}$$

where $Auth^n(sa_i)$ is the authority score for super-article sa_i in n^{th} iteration, $Hub^n(w_i)$ is the hub score for website w_i in n^{th} iteration, z is the number of websites that sa_i belongs to, and v is the number of super-articles that w_i contains. We define the similarity between two super-articles as the average of the similarity values among all their members, that is,

$$sim(sa_i, sa_j) = \frac{\sum_{a_m \in sa_i} \sum_{a_n \in sa_j} sim(a_m, a_n)}{|sa_i| \cdot |sa_j|}. \tag{7}$$

Accordingly, $msup(k, j)$, which means the support from super-article sa_k to super-article sa_j , is represented as

$$msup(k, i) = \frac{sim(sa_k, sa_i)}{\sum_{sa_j \in SA} sim(sa_k, sa_j)}, \tag{8}$$

and we define $msup(i, i) = 0$. Similar with TWEM, the final hub scores are combined with popularity to get the topical importance for websites. Note that the primary aim

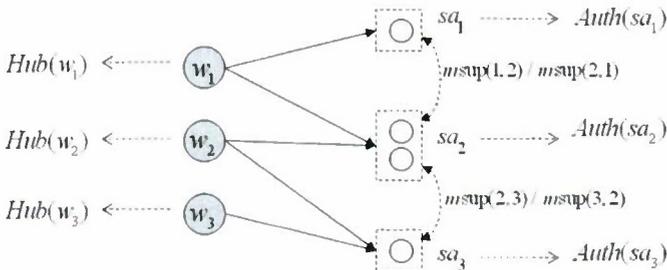


Fig. 4. The merge-TWEM model

of our models is to rank websites, though we make great efforts (e.g., consideration of article support and adoption of article merging) on modeling relations among news articles.

4 Experiments

In this section, we conduct experimental studies to evaluate the effectiveness of the TWEM and merge-TWEM models. Before going to the details, we first describe the dataset and evaluation method.

4.1 Dataset and Evaluation Method

There are no benchmark datasets for evaluating our proposed models. In our experiments, we select ten testing cases which have been hot news topics during the last two years. For each testing case, we submit a representative query to Google News and download the first 400 articles which form the article collection. The websites which are extracted from the URLs of the articles form the website collection. More details about this dataset can be found in Appendix 1.

Both TWEM and merge-TWEM are run on this dataset. For each topic, these two models output websites which have been ranked according to their topical importance. We evaluate the results using a manual-scoring strategy, which consists of four steps as follows.

Step 1: The websites in the results are divided into 4 groups by assessors according to their ranks and topical importance generated by our computation model. Each group is marked as “Very Important”, “Important”, “Unimportant” and “Very Unimportant”, respectively. Then each website gets an *importance level* accordingly. This level can be viewed as the “classification” result of the models.

Step 2: From each group, we select randomly 10 websites. These 40 selected websites are used as the evaluation samples.

Step 3: Three assessors browse the articles of each sample and assign it a score (between 0 and 4) independently. Then this sample is given another importance level according to the average of the three scores. This level is the ground-truth result for this sample.

Step 4: After comparing the importance levels assigned in Step 1 and Step 3, the *Evaluation Sample Precision* (ESP) can be calculated as the proportion of evaluation samples whose importance levels in Step 1 and Step 3 are identical to the total number of evaluation samples.

ESP measures whether TWEM or merge-TWEM can rank websites correctly on news topics. We use ESP as the evaluation metric in our experiments, and the overall performance is evaluated by averaging the individual ESP values over the 10 topics.

4.2 Performance Evaluation

The parameters are set in the following ways. The threshold in the merge-TWEM model is set to 0.9 since we want to exert strict restrictions on the merging operation. The controlling factor α is set experimentally. We tune it from 0 to 1.0 with 0.1 as the

step size, and Figure 5 shows the variance of performance for TWEM and merge-TWEM. We can see that both models perform best when α is equal to 0.3. Therefore, we set α to 0.3.

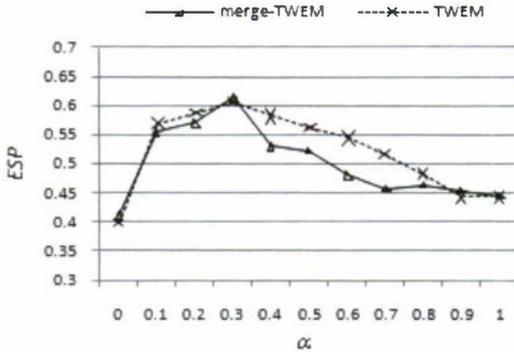


Fig. 5. ESP of TWEM and merge-TWEM as α varies

We compare the results of TWEM and merge-TWEM with that of Google News and Alexa. With Google News, we can only get ranks of news articles. For a testing topic t , we take the procedures in Figure 6 to generate the ranks of websites from Google News. This is actually a *Weighted Voting* strategy to rank websites. Alexa ranks have been widely used for website ranking and evaluation. For each testing topic, the websites in the dataset are ranked simply according to their Alexa ranks. The Google News and Alexa results are also evaluated with the method introduced in Section 4.1. Table 1 shows the experimental results of various methods.

In the table, we can see that both TWEM and merge-TWEM are able to achieve better performance than the baselines. TWEM outperforms Google News and Alexa

Step 1: A group of keywords which are representative of t are submitted to Google.

Step 2: The first 400 web pages returned by Google are downloaded. Websites are extracted from the URLs of these web pages.

Step 3: Each website is assigned a *Google score* which equals the weighted summation of the Google ranks of its web pages, that is

$$score(w) = \sum_{p_i \in w} c_i \cdot rank(p_i)$$

where p_i is i^{th} web page of the website w , $rank(p_i)$ is the Google rank of the page p_i , c_i is the coefficient for the rank of p_i .

Step 4: The websites are finally ranked according to their *Google scores*.

Fig. 6. Generation of Google ranks for websites

by 25.5% and 36.0% respectively, while the achievements obtained by merge-TWEM are 27.6% and 38.2%. In order to determine whether these improvements are statistically significant, we perform several single-tailed t-tests, and Table 2 gives the P-values of TWEM and merge-TWEM compared to Google News and Alexa. From this table, we find that either TWEM or merge-TWEM performs significantly better than the baselines at a 95% confidence level.

Table 1. Performance comparison between the models and the baselines

Methods	ESP
TWEM	0.6062
merge-TWEM	0.6162
Google News	0.4830
Alexa	0.4460

Table 2. P-values of the t-tests. (a) P-values of TWEM and merge-TWEM compared to Google News. (b) P-values of TWEM and merge-TWEM compared to Alexa.

(a)	
Methods	P-values
TWEM	0.0142
merge-TWEM	0.0110

(b)	
Methods	P-values
TWEM	0.0092
merge-TWEM	8.24e-4

Table 3. Performance comparison among TWEM, TWEM-S, TWEM-P, TWEM-S-P

Models	ESP
TWEM	0.6062
TWEM-S	0.4360
TWEM-P	0.4132
TWEM-S-P	0.3208

Moreover, we conduct comparison between the merge-TWEM and TWEM models. We observe from Table 1 that merge-TWEM has higher ESP than TWEM (0.6162 vs 0.6062). Also Table 2 shows that merge-TWEM has smaller P-values than TWEM in any cases. This proves that the article merging operation indeed takes effects in boosting the performance of TWEM. The merge-TWEM model, which considers the special features of news articles, can generate better ranking results and thus serve as a more effective solution to the NWE problem.

Finally, we provide a detailed view of the TWEM model. TWEM utilizes support among news articles and popularity of websites. We investigate the impact of these two factors. After excluding each of them from TWEM, we get two new models, i.e.,

TWEM-S and TWEM-P. When the two factors are both excluded, TWEM boils down to the basic HITS method, which is named as TWEM-S-P. These newly-derived models are run on the dataset and their results are evaluated in the same way as TWEM. Table 3 shows the ESP values for them. In the table, we find that TWEM performs better than all the other three models. Among the two factors, popularity of websites brings more significant improvements (0.4132 VS 0.6062). Also, consideration of support among news articles also improves the performance greatly (0.4360 VS 0.6062). Based on the above comparison, we conclude that these two factors play important roles in the performance of TWEM.

5 Conclusion and Future Work

In this paper, we study extensively the problem of News Website Evaluation. We propose two models, i.e., TWEM and merge-TWEM, to solve this problem. TWEM exploits fully the relations between websites and news articles to infer the importance scores of websites. Also, TWEM utilizes information from Alexa to represent popularity of websites. The merge-TWEM model improves TWEM by incorporating the article merging operation. The experiments show that both TWEM and merge-TWEM outperform the baseline systems significantly. In addition, the merge-TWEM model achieves better performance than TWEM, and is a more effective solution to the NWE problem.

In this paper, we mainly focus on the importance of websites on news topics. In our future work, we will consider extending the proposed models to other types of topics. Furthermore, our evaluation procedures are still based on human judgment. Therefore, we will also study more reasonable and objective evaluation methods.

Acknowledgments. This work was supported by National Natural Science Funding of China under Grant No. 90718022 and National 863 Project under Grant No. 2009AA01Z410.

References

1. Kleinberg, J.M.: Authoritative Sources in a Hyperlinked Environment. In: Proceedings of the 9th Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 668–677 (1998)
2. Yin, X., Han, J., Yu, P.S.: Truth Discovery with Multiple Conflicting Information Providers on the Web. *IEEE Transactions on Knowledge and Data Engineering* 20, 796–808 (2008)
3. Yin, X., Han, J., Yu, P.S.: Truth Discovery with Multiple Conflicting Information Providers on the Web. In: Proceedings of the 13th Annual International ACM SIGKDD Conference on Knowledge Discovery and Data Mining, pp. 1048–1052 (2007)
4. Liu, Y., Gao, B., Liu, T., Zhang, Y., Ma, Z., He, S., Li, H.: BrowseRank: Letting Web Users Vote for Page Importance. In: Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 451–458 (2008)
5. Zhu, G., Mishne, G.: Mining Rich Session Context to Improve Web Search. In: Proceedings of the 15th Annual International ACM SIGKDD Conference on Knowledge Discovery and Data Mining, pp. 1037–1046 (2009)
6. Dai, C., Lin, D., Bertino, E., Kantarcioglu, M.: Trust Evaluation of Data Provenance. CERIAS Tech. Report (2008)

7. Page, L.: PageRank: Bringing Order to the Web. In: Stanford Digital Libraries Working Paper (1997)
8. Haveliwala, T.H.: Topic-sensitive PageRank. In: Proceedings of the 11th International World Wide Web Conference, pp. 517–526 (2002)
9. Bharat, K., Henzinger, M.R.: Improved Algorithms for Topic Distillation in Hyperlinked Environments. In: Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 104–111 (1998)
10. Chakrabarti, S., Dom, B.E., Raghavan, P., Rajagopalan, S., Gibson, D., Kleinberg, J.M.: Automatic Resource Compilation by Analyzing Hyperlink Structure and Associated Text. In: Proceedings of the 7th International World Wide Web Conference, pp. 65–74 (1998)
11. Nie, L., Davison, B.D., Qi, X.: Topical Link Analysis for Web Search. In: Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 91–98 (2006)
12. Wang, C., Zhang, M., Ru, L., Ma, S.: Automatic Online News Topic Ranking Using Media Focus and User Attention Based on Aging Theory. In: Proceeding of the 17th ACM Conference on Information and Knowledge Management, pp. 1033–1042 (2008)
13. Fernandes, D., Moura, E.S., Ribeiro-Neto, B.: Computing Block Importance for Searching on Web Sites. In: Proceedings of the 16th ACM Conference on Information and Knowledge Management, pp. 165–174 (2007)
14. Dou, Z., Song, R., Nie, J., Wen, J.: Using Anchor Texts with Their Hyperlink Structure for Web Search. In: Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 227–234 (2009)
15. Guo, F., Liu, C., Kannan, A., Minka, T., Taylor, M., Wang, Y., Faloutsos, C.: Click Chain Model in Web Search. In: Proceedings of the 18th International World Wide Web Conference, pp. 11–20 (2009)
16. Song, G., Miao, Y., Yang, L., Li, C.: Topic-Based Computing Model for Web Page Popularity and Website Influence. In: Proceedings of the 22nd Australasian Joint Conference on Artificial Intelligence, pp. 210–219 (2009)

Appendix 1: Details about the Dataset

Topic No.	Brief Descriptions	Number of Websites	Number of Pages
1	An employee jumped from the fourth floor of a famous company's office building and died.	118	208
2	Insiders manipulated the stock price of JSSH, a bioengineering company in China.	42	54
3	The battery of Huntkey exploded in a foreign test, which was later reported widely.	39	64
4	Google China was punished by Chinese government for disseminating vulgar links and images.	188	365
5	A famous broadcaster in China was reported to be a spy. Finally, this proved to be a rumor.	207	367
6	A farmer claimed that he took photos of South China Tiger, a species which had been thought to be extinct.	132	361
7	In China, a girl in a TV show was found to look extremely like a super star, and received public attention	78	100
8	In Sep 2009, Lenovo again launched laptop computers specially designed for university students.	122	187
9	A new movie, Sophie's Revenge, was released in Sep 2009 and broke the box-office records in China.	81	96
10	Windows is about to be released in Oct 2009. There are already many comments and discussions about it.	96	143

A Statistical Interestingness Measures for XML Based Association Rules

Izwan Nizal Mohd Shaharance, Fedja Hadzic, and Tharam S. Dillon

Digital Ecosystem and Business Intelligence Institute, Curtin University of Technology,
Perth 6102, Australia

izwan.mohdshaharancee@postgrad.curtin.edu.au,
{f.hadzic, tharam.dillon}@cbs.curtin.edu.au

Abstract. Recently mining frequent substructures from XML data has gained a considerable amount of interest. Different methods have been proposed and examined for mining frequent patterns from XML documents efficiently and effectively. While many frequent XML patterns generated are useful and interesting, it is common that a large portion of them is not considered as interesting or significant for the application at hand. In this paper, we present a systematic approach to ascertain whether the discovered XML patterns are significant and not just coincidental associations, and provide a precise statistical approach to support this framework. The proposed strategy combines data mining and statistical measurement techniques to discard the non significant patterns. In this paper we considered the “Prions” database that describes the protein instances stored for Human Prions Protein. The proposed unified framework is applied on this dataset to demonstrate its effectiveness in assessing interestingness of discovered XML patterns by statistical means. When the dataset is used for classification/prediction purposes, the proposed approach will discard non significant XML patterns, without the cost of a reduction in the accuracy of the pattern set as a whole.

Keywords: data mining, interesting rules, statistical analysis, semi-structured data.

1 Introduction

Data mining or knowledge discovery from data (KDD) is known for its capabilities in extracting knowledge that is comprehensible, valid on tests and new data with some degree of certainty, potentially useful, actionable, and novel [1]. With the fast growth in the amount of electronic data such as Web pages and XML data, this offers a new dimension in pattern recognition and rules discovery. These electronic data are heterogeneous collection of ill-structured data that have no rigid structures, and often referred to as semi-structured data [2]. A well known data mining technique, namely association rule mining is widely used for discovering interesting associations and correlations between data elements in a diverse range of applications. While there are great achievements in discovering the association rules within the well-structured

(relational) data, still a number of works remain in preliminary stages for semi-structured data [3]. Since the introduction of the association rule mining problem by [4], substantial work has gone into various trends, including the development of efficient algorithms in finding the association [5-7] and measuring the interestingness of the association rules in structured data [8-14]. As the increase in data captured in semi structured format such XML begins to permeate many applications, association rule mining from the semi-structured data has become a new and interesting research area [15]. The general problems of association rule mining include the extraction of all the frequent itemsets from which association rules are formed. A rule is said to be interesting if they meet certain minimum support and confidence criteria [3]. The same holds for mining the frequent substructures in semi-structured data which comprise candidate substructure enumeration and frequency counting.

Works such as [2, 16-18] focus on developing algorithms to enable efficient and effective association rule mining from semi-structured data. While these frequent substructure mining techniques may discover an interesting association from a given dataset, the problem that remains is that they may only reflect aspects of the database being observed. As such, the patterns may not reflect the "real" significant associations between the underlying structures. This problem arises because some association rules are discovered due to pure coincidence resulting from certain randomness in the particular dataset being analyzed. Since the nature of data mining techniques is data driven, the patterns generated by these techniques must be validated by a statistical methodology for them to be useful in practice [19]. Statistics has previously addressed the issues of how to separate out the random effects to determine if the measured association (or difference in other areas) is significant [20]. Thus additional measures based on statistical independence and correlation analysis are needed to ensure that the results have a sound statistical basis and are not purely random coincidence.

Therefore, the motivation behind our proposed method is to investigate how data mining and statistical measurement techniques can be combined to arrive at more reliable and interesting set of rules. The focus of the work presented in this paper is to evaluate the frequent substructures extracted from XML documents and verify their significance using statistical analysis. In this paper we apply the IMB3 algorithm [21] to the Prions database in order to extract the frequently occurring substructures, while statistical analysis, namely Chi-Squared and Log-Linear have been utilized to ascertain the discovered substructures. In the next section, we explain the problem of discovering and ascertaining association rules from semi structured data. In Section III, we describe some related works in the area of frequent substructure mining and finding of significant patterns. We show experimental findings of significant substructures in Prions dataset in Section IV. Section V concludes the paper and explains our ongoing work in this field of study.

2 Problem Definition

This section starts by describing some necessary aspects of association rule mining in the context of XML document mining which will lay the ground work to define the problem of ascertaining patterns/association rules from semi-structured data. XML

document has a hierarchical document structure, where an XML element may contain further embedded elements, and these can be attached with a number of attributes. Elements that form sibling relationships may have ordering imposed on them. Each element of an XML document has *name* and *value*. Given such parallelisms, an XML document can therefore be modeled as a rooted labeled ordered tree, where a node in the tree corresponds to an XML element [15, 17]. If only structure is to be considered, then a node in the tree will only correspond to an element name. However, in the case of the current study we are interested in attribute names and the attribute values from a particular domain, and hence a node will correspond to an element name and value.

A tree can be denoted as $T(v_0, V, L, E)$, where:

- (1) $v_0 \in V$ is the *root* vertex;
- (2) V is the set of *vertices* or *nodes*;
- (3) L is the set of *labels* of vertices, for any vertex $v \in V$, $L(v)$ denotes the label of v ; and
- (4) $E = \{(x, y) | x, y \in V\}$ is the set of *edges* in the tree.

The main problem in association mining from semi-structured documents such as XML, is that of frequent pattern discovery, where a pattern corresponds to a subtree in this case, and a transaction to a fragment of the database tree whereby an independent instance is described. This problem is more complex than in traditional frequent pattern mining from relational data because structural relationships need to be taken into account. It is known as the **frequent subtree mining** problem, and can be generally stated as: given a tree database T and minimum support threshold (σ), find all subtrees that occur at least σ times in T .

Furthermore, depending on the domain of interest and the task that is to be accomplished in a particular application, different types of subtrees can be mined using different support definitions. For an overview of existing subtree types and support definitions and their usage implications for general knowledge analysis tasks please refer to [22]. Many frequent subtree mining algorithms have been developed to date, and for an extensive overview of the current state-of-the-art in the field, including comparisons of different approaches highlighting their advantages/disadvantages, we refer the interested reader to [3, 15].

Due to the nature of the domain considered and the data used in this paper we focus on ordered induced subtrees and the transaction based support definition is used. These can be formally defined as follows:

Definition 1. Given a tree $S = (v_0_S, V_S, L_S, E_S)$ and tree $T = (v_0_T, V_T, L_T, E_T)$, S is an **ordered induced subtree** of T , iff (1) $V_S \subseteq V_T$; (2) $L_S \subseteq L_T$ and $L_S(v) = L_T(v)$; (3) $E_S \subseteq E_T$; and (4) the left to right ordering of sibling nodes in the original tree is preserved.

When using the **transaction-based support (TS)** definition, the transactional support (σ) of a subtree t , denoted as $\sigma_{tr}(t)$ in a tree database T_{db} is equal to the number of transactions in T_{db} that contain at least one occurrence of subtree t .

Definition 2. Let the notation $t \prec k$, denote the support of subtree t by transaction k , then for **TS**, $t \prec k = 1$ whenever k contains at least one occurrence of t , and 0 otherwise. Suppose that there are N transactions k_1 to k_N of tree in T_{db} , the $\sigma_{tr}(t)$ in T_{db} is defined as:

$$\sum_{i=1}^N t < k_i \quad (1)$$

Hence, in our current work we focus on ascertaining the interestingness of discovered ordered induced subtree patterns that have been extracted from a tree-structured database (XML), and that satisfy the minimum transaction-based support threshold.

Let us denote the set of these frequent subtree patterns as SF . Please note that the patterns from SF have not been assigned a particular class label to be used for a prediction/classification task, and as such simply reflect the frequently occurring associations, that may not necessarily have a sound statistical basis. Hence, in the first problem setting our aim is to reduce the SF , by filtering out the patterns that are not statistically significant with respect to the statistical measures used.

In the second problem setting, one of the attributes from the data is considered as a class to be predicted for classification task purposes. Hence, we only consider those patterns from SF , that contain this class attribute, as they will represent the set of values that frequently occur together when a particular class value is present. Hence, as such these patterns can be seen to have predictive power and can be evaluated for their accuracy on correctly predicting the class value from the trained data and unseen data. In addition to predictive accuracy, simple rules are preferred as they are easier to comprehend and are expected to perform better on unseen data since they are more general. Hence, when in the process of optimizing a rule set, a trade-off needs to be made between several factors and the common ones are:

- *Misclassification rate (MR)* – number of incorrectly classified instances
- *Coverage rate (CR)* - number of captured instances
- *Generalization power (GP)* – capability of correctly classifying future instances

When optimizing the rule set, the MR should be minimized while the CR should be maximized. GP is achieved by simplifying the rules in terms of overall rule set size and the number of attribute constraints in the rule. The trade-off occurs especially when the data set is characterized by continuous attributes where a valid attribute range constraint needs to be determined for a particular rule. Increasing the range constraint usually leads to the increase in CR of that rule but at the cost of an increase in MR of that rule. Similarly, if the rules are too general, they may lack the specificity to distinguish some domain characteristics and hence the MR would increase. Generally speaking, an optimized rule set should be either more accurate than the original rule set and/or the balance between the trade-off factors should be much greater. For example, if there are many rules with small CR but very low MR , a rule set with a significantly smaller number of rules may be preferred even at the cost of an increase in MR .

Since the number of patterns/association rules generated through frequent subtree mining can be quite large, their usefulness for classification/prediction task may be limited unless they are significantly reduced in size and number. While their MR may be small, their GP is likely to be poor as all frequent patterns are considered, that can be insignificant, redundant and unnecessarily complex. Hence in the second problem considered in this paper, we aim to apply a variety of statistical/heuristic methods to reduce the pattern/rule set size and simplify individual rules.

Let us denote the subtree patterns from the frequent subtree set SF that have a class label (value), as SFC . The problem considered in the second setting can be stated as. Given SFC with accuracy ac , reduce SFC into SFC' such that SFC' has accuracy $\geq (ac - \epsilon)$, such that ϵ is an arbitrary user defined small value (ϵ is used to reflect the noise that is often present in real world data).

3 Related Works

Our work in this paper focused on ascertaining the XML rules discovered from an XML-enabled association rule framework. [17] have initiated this framework which resulted in a more flexible and powerful representations of both simple and complex structured association relationships inherent in XML documents. There has been an active development of frequent subtree mining algorithms [16, 18, 21, 23-25]. For a more detailed description of the existing approaches and latest development on these algorithms please refer to [3, 15]. Currently there has been limited works in rule evaluation phase of semi-structured rules. Many of the well developed rule interestingness measures are in structured data and they have had great success in evaluating rule interestingness as discussed in [12]. Initial work on evaluating the discovered patterns based on statistical significant are [13, 26-28] but these are limited to structural data. The existence of vast well developed measuring techniques to evaluate interestingness of rules from relational data, offers great opportunities in adapting these techniques for verifying significant substructures from semi-structure data. The applicability of these interestingness measures needs to be explored in context of frequent substructure mining, where necessary adjustments and extensions need to take place to ascertain the validity of the methods in presence of more complex structural aspects in the data, which often need to be preserved in the rules.

One line of work in focusing on more interesting substructure patterns is in reducing the patterns and the application of plausible constraints techniques. The problem of mining mutually dependent ordered subtrees has been addressed in [29]. The proposed algorithm utilizes the hyperclique method [30] in the tree mining context so that all the components of a subtree are highly correlated together. These hyperclique subtree patterns are discovered using a *h-confidence* measure which is the minimum probability of an item from a pattern in one transaction implying the presence of all other items in the same transaction. Hence, the extracted hyperclique subtree patterns will satisfy the minimum *h-confidence* threshold. The work done in [31] uses the method proposed for database compression in regards to item set mining in [32] to demonstrate how the same minimum description length principle can yield good results for sequential and tree-structured data. Another notable work presented in [33] extends the idea of the item constraint [34] to that of node-inclusion constraint in subtrees. In addition to that, [35] proposed the application of monotone constraint namely anti-monotone, monotone convertible and succinct in frequent subtree mining. Such an opportunistic pruning strategy is used to mine frequent subtrees under the defined constraints. An approach for mining of frequent subtrees where the distance between the nodes is used as additional grouping criterion has been presented in [36]. [37] proposed and demonstrated an efficient ways to discover interesting association

rules from dynamic XML documents. The work done in [37] mainly motivated by the facts that the XML document's content and /or structure are always fluctuates.

Besides the aforementioned constraint-based techniques, to our knowledge we found limited works on verifying the significance of discovered frequent substructures. The frequent occurring substructure discovered from frequent substructure mining algorithm commonly offers a complete pattern set and is too numerous to be utilized efficiently and effectively for the application at hand [9, 38]. [39] proposed and developed an application of statistical hypothesis testing to re-rank the significant frequent subtrees. This approach ranks the significant patterns according to *P-values* obtained from the *Fisher's Exact* test of significance. The significant patterns were then used for Glycan classifications problems. Recently [38], proposed a mining framework called LEAP (Descending Leap Mine) in checking and mining a significant frequent subgraph which will help in discarding redundant frequent subgraphs. For a predefined class label in XML documents, an efficient XRules classifier have been developed by [40]. This approach offers promising results in terms of the structural classifier for semi-structured data.

In this work we employed the IMB3 miner algorithm for mining ordered embedded subtrees. While these algorithms, offer some constraints in discovering strong patterns/rules, many misleading, uninteresting and insignificant rules in that domains may still be produced [1]. The problem arises because some association rules are discovered due to pure coincidence resulting from certain randomness in the particular dataset being analyzed. Statistics has previously addressed the issues of how to separate out the random effects to determine if the measured association (or difference in other areas) is significant [20]. Thus additional measures based on statistical independence and correlation analysis are needed to ensure that the results have a sound statistical basis and are not purely random coincidence.

A common multivariate statistical analysis is the association analysis problem [20]. For associations between categorical variables there are several inferential methods involved. Chi-Squared analysis is often used to measure the difference between observed and expected frequencies. The significance used of the Chi-Squared statistics is for hypothesis testing in tests of independence. In addition to that the Log-Linear analysis offers a unique feature in capturing interrelationship among data items [41].

4 Experimental Results

The evaluation of the unification framework is performed using the Prions database which is a type of infectious agent. Prions are abnormally structured forms of host protein, which are able to convert normal molecules of protein into abnormally structured form. Prions dataset describes Protein Ontology database for Human Prions proteins in XML format [42]. It consists of 17348 protein sequences. The XML tags and values are first mapped to integer indexes similar to the format used in [21] and [25]. Representing label as integer instead of a string label has considerable performance and space advantages [21]. In this section, we first show the generated patterns obtained from frequent subtree mining approach, namely IMB3 algorithm in Section 4.1. Then we apply the two prominent statistical measurement techniques

namely Chi-Squared analysis and Log-Linear analysis in measuring the significance of the discovered frequent patterns in Section 4.2. In Section 4.3, we consider the Prions protein database in a classification/prediction problem setting. We have labeled the protein instances as either referring to Human's or Animal's protein. We then verified the extracted patterns using the statistical analysis.

4.1 Extracted Frequent Patterns

The discovery of structural patterns by matching data representation structures is essential for analysis and understanding of data. If a structural pattern occurs frequently, it is ought to be important in some way. On the other hand, infrequent patterns may also provide meaningful information [42]. Thus to extract meaningful information from XML data we need to mine structural patterns. In discovering the frequent patterns from Prions dataset we apply the IMB3 algorithm. There are a total of 27 occurring patterns discovered by IMB3 algorithm. The minimum support value used was 10 % and we managed to discover subtree patterns with the largest ones consisting of 5 nodes. Table 1 shows several examples of patterns discovered.

Table 1. Examples of Several Patterns Discovered Based on Frequent Tree Mining Technique

Patterns #	Patterns	# of Occurrences
1	ATOMChain(A) Element(C)	3957
2	ATOMChain(A) ATOMResidual(TYR) Occupancy (1)	1743
3	ATOMChain(A) Occupancy(1) Temperature(0) Element(C)	3805

Pattern number 1 shows an association between *ATOMChain(A)* with *Element(C)* and this pattern was discovered 3957 times. Here the *ATOMChain* with value *A* associates to *Elements* with value *C*. The patterns discovered by the IMB3 algorithm can aid in discovering potentially useful pattern structures in Protein Ontology datasets, which makes it useful for comparison of protein datasets taken across protein families and species and helps in discovering interesting similarities and differences. However, the question still remains whether these patterns are discovered due to pure coincidence resulting from certain randomness in the particular dataset being analyzed. Furthermore, they are often quite large in number, which can degrade the analysis procedure, and hence in the next section we measure the statistical significance of the discovered patterns, in order to remove any non-significant patterns.

4.2 Frequent Patterns Significant Test

Statistical analysis approaches, namely Chi-Squared and Log-Linear analysis were employed in order to determine the usefulness of frequent rules obtained. The results from Chi-Squared analysis are discussed first.

Table 2. Patterns Verification Based on Chi-Squared Analysis

Node Name		Sig. Att. Value
ATOMResidual(TYR)	Occupancy(1)	Not Sig.
Occupancy(1)	Temperature(0)	Not Sig.
ATOMChain(A)	Occupancy(1)	Not Sig.
ATOMChain(A)	Element(C)	Sig.
ATOMChain(A)	Element(H)	Sig.
Temperature(0)	Element(C)	Sig.
Temperature(0)	Element(H)	Sig.
ATOMChain(A)	ATOMResidual(TYR)	Sig.
ProteinOntologyID(3)	Occupancy(1)	Not Sig.
ProteinOntologyID(3)	Element(C)	Sig.
ATOMChain(A)	Temperature(0)	Sig.
Occupancy(1)	Temperature(1)	Not Sig.
Occupancy(1)	Element(N)	Not Sig.
Occupancy(1)	Element(C)	Not Sig.
Occupancy(1)	Element(O)	Not Sig.
Occupancy(1)	Element(H)	Not Sig.

Table 2 shows that, there are 16 association relationships among structures-values items discovered using the IMB3 algorithm. Based on Chi-Squared analysis, 7 out of 16 relationships are significant. Table 3 shows 11 patterns with more than two nodes. We apply the Log-Linear analysis in examining the association between these nodes. Only one pattern out of 11 patterns is accepted as a significant pattern based on this analysis. Based on the Log-Linear analysis, we can conclude that, there is significant association between *ATOMChain(A)*, *Temperature(0)* and *Element(H)*.

Table 3. Patterns Verification Based on Log-Linear Analysis

Node Name				Sig. Att. Value
ATOMChain(A)	ATOMResidue(TYR)	Occupancy(1)		Not Sig.
ATOMChain(A)	Occupancy(1)	Temperature(0)		Not Sig.
ATOMChain(A)	Occupancy(1)	Element(C)		Not Sig.
ProteinOnto(3)	Occupancy(1)	Element(C)		Not Sig.
ATOMChain(A)	Occupancy(1)	Element(H)		Not Sig.
Occupancy(1)	Temperature(0)	Element(C)		Not Sig.
ATOMChain(A)	Temperature(0)	Element(C)		Not Sig.
Occupancy(1)	Temperature(0)	Element(H)		Not Sig.
ATOMChain(A)	Temperature(0)	Element(H)		Sig.
ATOMChain(A)	Occupancy(1)	Temperature(0)	Element(C)	Not Sig.
ATOMChain(A)	Occupancy(1)	Temperature(0)	Element(H)	Not Sig.

4.3 Prions as a Classification Problem

As in our previous work [11], the unification framework involves several steps in ascertaining the rules discovered from association rules mining process. For Prions dataset, the similar steps were followed. We defined a new variable (target variable) identified as Human Protein or Animal Protein class. This new variable was derived from ProteinOntologyID and SuperFamily variables. Hence, we have excluded the

ProteinOntologyID and SuperFamily variables from the dataset to be considered in this task. Thus in this classification problem we have chosen the target variable (i.e. Human or Animal's Protein) as the right hand side/consequence of the association rules.

In this experiment, we divided the Prions dataset into 60% of training set and 40% of testing set. Then we apply the preprocessing techniques including the missing values removal and discretization of attributes with continuous data. The equal depth binning approach method was selected as this approached offered a better result as discussed in [11]. The determination of relevant attributes with respect to being able to predict the target attributes is shown in Table 4. This is based on Symmetrical Tau [43] and Mutual Information [12] techniques. As discussed in [11], the Symmetrical Tau (ST) approach offers better output in discriminating criterions for class to be predicted in comparison to Mutual Information (MI), as it does not favor multi-valued attributes. The attributes with ST values that are respectively lower than other attribute's ST values, are considered as irrelevant for the task. The significant difference was considered to occur at the position where that attribute's ST value is less than half of the previous attribute's ST value in the ranking. Hence for this dataset, attributes 'Occupancy' and 'Y' were considered as irrelevant for the prediction task and were removed.

Table 4. Comparison between ST and MI for Prions Dataset

Variables	ST Values	Variables	MI Values
ATOMChain	0.2088	ATOMChain	0.2605
Temperature	0.1230	Z	0.1610
Z	0.0812	Temperature	0.1526
ATOMid	0.0407	ATOMResSeqNum	0.1053
ATOMResSeqNum	0.0280	ATOMid	0.0721
X	0.0256	X	0.0549
Element	0.0153	Atom	0.0238
Atom	0.0109	ATOMResidue	0.0187
ATOMResidue	0.0082	Element	0.0162
Y	0.0029	Y	0.0048
Occupancy	0.0001	Occupancy	0.0000

Table 5. Examples of Prions Rules

Set Size	Confidence	Support	Count	Rules
2	75.32	8.97	934	X(g) ==> Class (Animal)
4	61.71	6.66	693	X(d) & Z(b) & ATOMChain(A) ==> Class (Human)

Next, the rules are then generated based on the minimum support and confidence framework of 5% and 60% respectively. Table 5 shows examples of the generated rules. The discovered rules are then ascertained with statistical techniques namely Chi Squared [20] and Logistics Regression [20]. Based on these statistical analyses we found that only variables *ATOMChain*, *ATOMResidual*, *ATOMResSeqNum*, *X* and *Z* were significant contributors towards target variable of class Human or Animal.

Additional constraint measurement techniques were applied in order to discard the existence of redundant rules [11, 13]. The combination of these rule ascertaining strategies will facilitate the association rule mining framework to determine the right and high quality rules. These rules will have a sound statistical basis and we can be more confident that they reflect the real world situation.

In Table 6 we show the progressive difference in the number of rules generated as statistical analysis and redundancy checks are being utilized. We also show the respective classification (% of correctly classified instances from the training set) and predictive accuracy (% of correctly classified instances from the training set) of those rule sets. Upon a removal of 73% rules, we found that both classification and predictive accuracies have increased by more than 5%. This demonstrates the importance of ascertaining the association rules by statistical analysis and redundancy check, as in this particular scenario the simplified rule set is more general and performs better on unseen data.

The combination of statistical significance analysis and redundant analysis provided proper ways in discarding non significant rules, which is a significant reduction in the overall complexity of the rule set. From Table 6 we can also see that this great reduction of rules was not at a cost of a reduction in accuracy, as it in fact increased for the Prions dataset in classifying and predicting the protein classes.

Table 6. Rules Accuracy for Prions Data

*Dataset Description	Rule #	Type of Analysis	Accuracy	
			Classification	Prediction
Train : 10407 records	42	Initial rules	74.36%	75.00%
Test : 6938 records	11	Statistical Analysis / Redundancy Check	79.97%	80.37%

* Two records with missing values were discarded.

5 Conclusions and Future Works

This was our preliminary work towards the combination of data mining and statistical techniques in ascertaining the rules/patterns from semi-structured data. The combination of the approaches used in this method demonstrated a number of ways for ascertaining the significant patterns obtained using frequent subtree mining approaches. In this paper we employed statistical analysis that provides some control in lowering the risk of discovering a pattern that is false and spurious. In our future work we aim to test the approach using tree-structured data of various characteristics and complexities.

References

1. Han, J., Kamber, M.: Data mining: concepts and techniques. Morgan Kaufmann, San Francisco (2001)
2. Zhang, J., Ling, T.W., Bruckner, R.M., Tjoa, A.M., Liu, H.: On Efficient and Effective Association Rule Mining from XML Data. In: Proceedings of the 15th Int. Conf. Database and Expert Systems Applications, Zaragoza, Spain, pp. 497–507 (2004)
3. Chi, Y., Muntz, R.R., Nijssen, S., Kok, J.N.: Frequent Subtree Mining - An Overview. *Fundamenta Informaticae* 661, 61–198 (2005)

4. Agrawal, R., Imieliski, T., Swami, A.: Mining association rules between sets of items in large databases. *ACM SIGMOD Rec.* 22, 207–216 (1993)
5. Aggarwal, C.C., Yu, P.S.: A new framework for itemset generation. In: *Proceedings of the 17th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, pp. 18–24. ACM, Washington (1998)
6. Agrawal, R., Srikant, R.: *Fast Algorithms for Mining Association Rules in Large Databases*. In: *Proceedings of the 20th Int. Conf. on Very Large Data Bases*, Santiago, Chile (1994)
7. Toivonen, H.: Sampling Large Databases for Association Rules. In: *Proceedings of the 20th Int. Conf. on Very Large Data Bases*, Mumbai, India, pp. 134–145 (1996)
8. Bayardo, R., Agrawal, R., Gunopulos, D.: Constraint-Based Rule Mining in Large, Dense Databases. *J. Data Mining and Knowledge Discovery* 4, 217–240 (2000)
9. Lavrač, N., Flach, P., Zupan, B.: Rule Evaluation Measures: A Unifying View. In: *Džeroski, S., Flach, P.A. (eds.) ILP 1999. LNCS (LNAI), vol. 1634*, pp. 174–185. Springer, Heidelberg (1999)
10. Lenca, P., Meyer, P., Vaillant, B., Lallich, S.: On selecting interestingness measures for association rules: User oriented description and multiple criteria decision aid. *European Journal of Operational Research* 184, 610–626 (2008)
11. Shaharane, I.N.M., Hadzic, F., Dillon, T.: Interestingness of Association Rules Using Symmetrical Tau and Logistic Regression. In: *Nicholson, A., Li, X. (eds.) AI 2009. LNCS (LNAI), vol. 5866*, pp. 422–431. Springer, Heidelberg (2009)
12. Tan, P.N., Kumar, V., Srivastava, J.: Selecting the right interestingness measure for association patterns. In: *Proceedings of the 8th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, pp. 32–41. ACM, Alberta (2002)
13. Webb, G.I.: *Discovering Significant Patterns*. In: *Machine Learning*, pp. 1–33. Springer, Heidelberg (2007)
14. Yun, H., Ha, D., Hwang, B., Ho Ryu, K.: Mining association rules on significant rare data using relative support. *J. Systems and Software* 67, 181–191 (2003)
15. Tan, H., Hadzic, F., Dillon, T.S., Chang, E.: State of the art of data mining of tree structured information. *Int. Journal of Computer Systems Science and Engineering* 23 (2008)
16. Asai, T., Abe, K., Kawasoe, S., Arimura, H., Sakamoto, H., Arikawa, S.: Efficient Substructure Discovery from Large Semi-structured Data. In: *Proc. of the 2nd SIAM Int. Conf. on Data Mining (SIAM 2002)*, pp. 158–174 (2002)
17. Feng, L., Dillon, T., Weigand, H., Chang, E.: An XML-Enabled Association Rule Framework. *Database and Expert Systems Applications*, 88–97 (2003)
18. Tan, H., Hadzic, F., Dillon, T.S., Chang, E., Feng, L.: Tree model guided candidate generation for mining frequent subtrees from XML documents. *ACM Trans. Knowl. Discov. Data* 2, 1–43 (2008)
19. Goodman, A., Kamath, C., Kumar, V.: Data Analysis in the 21st Century. *Stat. Anal. Data Mining* 1, 1–3 (2008)
20. Agresti, A.: *An Intro. to Categorical Data Analysis*. Wiley Interscience, New Jersey (2007)
21. Tan, H., Dillon, T., Hadzic, F., Chang, E., Feng, L.: IMB3-Miner: Mining Induced/Embedded Subtrees by Constraining the Level of Embedding. In: *Proceedings of the 8th Pacific-Asia Conference on Knowl. Discovery and Data Mining*, pp. 450–461 (2006)
22. Fedja, H., Tharam, S.D., Elizabeth, C.: Knowledge Analysis with Tree Patterns. In: *Proceedings of the 41st Annual Hawaii Int. Conf. on System Sciences*. IEEE, Los Alamitos (2008)
23. Fedja, H., Henry, T., Tharam, S.D.: U3 - Mining Unordered Embedded Subtrees Using TMG Candidate Generation. In: *The 1st ACM Int. Conf. on Web Search and Data Mining*, California, USA (2008)

24. Tan, H., Dillon, T., Hadzic, F., Chang, E., Feng, L.: MB3-Miner: efficiently mining eMBEDded subTREES using Tree Model Guided candidate generation. In: Proceedings of the 1st Int. Workshop on Mining Complex Data 2005, Texas, USA (2005)
25. Zaki, M.J.: Efficiently Mining Frequent Trees in a Forest: Algorithms and Applications. *IEEE Transactions on Knowledge and Data Engineering* 17, 1021–1035 (2005)
26. Aumann, Y., Lindell, Y.: A Statistical Theory for Quantitative Association Rules. *J. Intell. Inf. Syst.* 20, 255–283 (2003)
27. Meggido, N., Srikant, R.: Discovering Predictive Association Rules. In: 4th International Conference on Knowledge Discovery in Databases and Data Mining, pp. 274–278 (1998)
28. Webb, G.I.: Preliminary investigations into statistically valid exploratory rule discovery. In: Simoff, S.J., Williams, G.J., Hegland, M. (eds.) *AusDM 2003*. Sydney, pp. 1–9 (2003)
29. Ozaki, T., Ohkawa, T.: Mining Mutually Dependent Ordered Subtrees in Tree Databases. In: *New Frontiers in Applied Data Mining: PAKDD 2008 Int. Workshops, Japan, Revised Selected Papers*, pp. 75–86. Springer, Heidelberg (2009)
30. Hui, X., Pang-Ning, T., Vipin, K.: Hyperrelique pattern discovery. *Data Min. Knowl. Discov.* 13, 219–242 (2006)
31. Bathoorn, R., Koopman, A., Siebes, A.: Reducing the Frequent Pattern Set. In: Proceedings of the 6th IEEE International Conference on Data Mining – Workshops, pp. 55–59 (2006)
32. Siebes, A., Vreeken, J., Leeuwen, M.V.: Item Sets That Compress. In: Proceedings of the SIAM Conference on Data Mining, Maryland, USA, pp. 393–404 (2006)
33. Nakamura, A., Kudo, M.: Mining Frequent Trees with Node-Inclusion Constraints. In: Ho, T.-B., Cheung, D., Liu, H. (eds.) *PAKDD 2005*. LNCS (LNAI), vol. 3518, pp. 850–860. Springer, Heidelberg (2005)
34. Srikant, R., Vu, Q., Agrawal, R.: Mining Association Rules with Item Constraints. In: 3rd Int. Conf. on Knowledge Discovery in Databases and Data Mining, Newport Beach, California, pp. 67–73 (1997)
35. Knijf, J.D., Feelders, A.J.: Monotone Constraints in Frequent Tree Mining. In: Poel, M., Nijholt, A. (eds.) *BENELEARN*, Enschede, The Netherlands, pp. 13–20 (2005)
36. Fedja, H., Henry, T., Tharam, D.: Mining Unordered Distance-Constrained Embedded Subtrees. In: Boulicaut, J.-F., Berthold, M.R., Horváth, T. (eds.) *DS 2008*. LNCS (LNAI), vol. 5255, pp. 272–283. Springer, Heidelberg (2008)
37. Rusu, L.I., Rahayu, W., Taniar, D.: Extracting Variable Knowledge from Multiversioned XML Documents. In: 6th IEEE International Conference on Data Mining - Workshops (ICDMW 2006), pp. 70–74 (2006)
38. Yan, X., Cheng, H., Han, J., Yu, P.S.: Mining significant graph patterns by leap search. In: *SIGMOD Conference*, Canada, pp. 433–444 (2008)
39. Hashimoto, K., Takigawa, I., Shiga, M., Kanehisa, M., Mamitsuka, H.: Mining significant tree patterns in carbohydrate sugar chains. *Bioinformatics* 24, 167–173 (2008)
40. Zaki, M.J., Aggarwal, C.C.: XRules: an effective structural classifier for XML data. In: *SIGKDD 2003*, Washington, DC (2003)
41. Wu, X., Barbar, D., Ye, Y.: Screening and interpreting multi-item associations based on log-linear modeling. In: Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM Press, Washington (2003)
42. Fedja, H., Tharam, S.D., Amandeep, S.S., Elizabeth, C., Henry, T.: Mining Substructures in Protein Data. In: Proceedings of the 6th IEEE Int. Conf. on Data Mining-Workshops (2006)
43. Zhou, X.J., Dillon, T.S.: A statistical-heuristic feature selection criterion for decision tree induction. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (1991)

Toward Improving Re-coloring Based Clustering with Graph b-Coloring

Hiroki Ogino and Tetsuya Yoshida

Graduate School of Information Science and Technology, Hokkaido University
N-14 W-9, Sapporo 060-0814, Japan
{hiroki,yoshida}@meme.hokudai.ac.jp

Abstract. This paper proposes an approach toward improving re-coloring based clustering with graph b-coloring. Previous b-coloring based clustering algorithm did not consider the quality of clusters. Although a greedy re-coloring algorithm was proposed, it was still restrictive in terms of the explored search space due to its greedy and sequential re-coloring process. We aim at overcoming the limitations by enlarging the search space for re-coloring, while guaranteeing b-coloring properties. A best first re-coloring algorithm is proposed to realize non-greedy search for the admissible colors of vertices. A color exchange algorithm is proposed to remedy the problem in sequential re-coloring. These algorithms are orthogonal with respect to the re-colored vertices and thus can be utilized in conjunction. Preliminary evaluations are conducted over several benchmark datasets, and the results are encouraging.

1 Introduction

When the dissimilarities among data items are specified, the entire data items can be represented as a graph structure, where each data item is mapped to a vertex and the vertices are connected by edges with the corresponding dissimilarities. Several graph-based clustering methods have been proposed [7,9,16]. Recently, [12] proposed the notion of b-coloring of undirected graphs. A graph b-coloring is a vertex coloring, and it satisfies the following constraints: (i) adjacent vertices have different colors, (ii) in each color, at least one vertex is adjacent to all the other colors. Based on this, [5] proposed a clustering method, but it did not consider the quality of clusters. Although a re-coloring algorithm was proposed to reflect the quality of clusters [6], it was still restrictive in terms of the explored search space due to its greedy and sequential process.

This paper proposes an approach toward improving re-coloring based clustering with graph b-coloring. The vertices in a graph are divided into two disjoint subsets based on the property of b-coloring. A best first re-coloring algorithm is proposed to realize non-greedy search for the admissible colors of vertices in one subset. The constraint (i) can make it impossible to re-color vertices in sequential approach. A color exchange algorithm is proposed so that this problem can be resolved. Both algorithms enlarge the search space for re-coloring, and re-color the vertices to improve the quality of clusters. Since these algorithms are orthogonal with respect to the re-colored vertices, they can be utilized in conjunction. Preliminary evaluations are conducted over several UCI datasets. The results are encouraging for pursuing this line of research, especially with respect to the ground truth micro-averaged precision.

1.1 Related Work

In general, clustering methods are divided into hierarchieal methods and partitioning methods [13]. Hierarchieal methods construet a cluster hierarchy, or a tree of elusters (called a dendrogram), where leaves correspond to data items and internal nodes to nested elusters of various sizes [8]. On the other hand, partitioning methods return a single partition of the entire data under a fixed parameters (number of clusters, thresholds, etc.). Each cluster can be represented by its centroid [10] or by one of its objects loeated around its center [15].

Until now various clustering methods have been proposed based on graph-theoretic concepts. In one approach, a partition of data items is obtained by removing edges and dividing the graph into several disconnected components. For instance, in spectral elustering, removal of edges are conceived in terms of the minimum cut of the graph, and eigenvectors of the (normalized) graph Laplaecian is utilized [16]. Other approaches utilize graph coloring techniques [4].

As for the coloring based approach, a hierarchieal agglomerative elustering method was proposed in [7]. It conduets a 2-coloring of vertices in order to find out a maximum spanning tree of a graph. In [9], partitioning of data items into elusters is conceived in terms of the minimal coloring of a graph. Our approach is yet another graph based partitioning method based on vertex coloring of a graph.

Section 2 describes an overview of b-coloring based elustering and points out some issues. The details of our proposal is presented in Section 3. Preliminary evaluations are reported in Section 4 and the results are diseussed. Section 5 describes concluding remarks and indicates future direCTIONS.

2 b-Coloring Based Clustering

2.1 Preliminaries

We use a bold capital letter to denote a set of objects. For a set V , $|V|$ represents its eardinality. A graph $G(V, E)$ consists of a set of vertices V and a set of edges E over $V \times V$. We assume that $G(V, E)$ is an undirected, simple graph without self-loop. The symbol Δ denotes the maximum degree in a graph [4].

Suppose data items are clustered or grouped into a partition $P = \{C_1, C_2, \dots, C_k\}$, where

C_i stands for a group (cluster) of data items. Since each cluster is represented as a color in our approach, we abuse the symbol P to represent both the set of clusters and the set of colors in a graph.

Table 1. Notations

symbol	description
n	the number of vertices
m	the number of edges
Δ	maximum degree of a graph
P	the set of colors in a graph
$c(v_i)$	the color of vertex v_i
$N(v_i)$	ncighboring vertices of vertex v_i
$N_c(v_i)$	neighboring colors of vertex v_i
$C_p(v_i)$	admissible colors for vertex v_i
$d(v_i, v_j)$	dissimilarity between v_i and v_j
$d_a(v, C_i)$	average dissimilarity between v and C_i

For a graph $G(\mathbf{V}, \mathbf{E})$, we define several functions over the vertices \mathbf{V} in G . A function $N(v)$ returns the set of vertices adjacent to the vertex v . A function $c(v)$ returns the color of v in G , and a function $N_c(v)$ returns the set of neighboring colors to v . A function $C_p(v)$ returns the set of admissible colors for v , i.e., the colors which are different from $N_c(v)$. Note that $C_p(v)$ contains the original color $c(v)$ of v .

It is assumed that a dissimilarity function $d: \mathbf{V} \times \mathbf{V} \rightarrow \mathcal{R}^+$ is specified for data items \mathbf{V} . For instance, $d(v_i, v_j)$ returns the dissimilarity between the pair of vertices v_i and v_j . For $\forall v \in \mathbf{V}, \forall C_i \in \mathbf{P}$, an average dissimilarity between v and C_i is defined as

$$d_a(v, C_i) = \frac{1}{|C_i|} \sum_{v_p \in C_i} d(v, v_p) \tag{1}$$

where $|C_i|$ denotes the size of cluster C_i .

The above notations are summarized in Table 1.

2.2 A Validation Index for Clustering

The objective of data clustering is to find out a partition with large *intra-cluster cohesion* and *inter-cluster separation* [13]. Various validation indices for clustering have been proposed [2]. Among them, we utilize an index called **generalized Dunn's index** $Dunn_G$. $Dunn_G$ is designed to offer a compromise between the *inter-cluster separation* and the *intra-cluster cohesion*.

For any $C_h \in \mathbf{P}$, an average within-cluster dissimilarity is defined as

$$S_a(C_h) = \frac{1}{|C_h|(|C_h| - 1)} \sum_{v \in C_h} \sum_{v' \in C_h} d(v, v') \tag{2}$$

For any pair of clusters $C_i, C_j \in \mathbf{P}$, an average between-cluster dissimilarity is defined as

$$d_a(C_i, C_j) = \frac{1}{|C_i||C_j|} \sum_{v \in C_i} \sum_{v' \in C_j} d(v, v') \tag{3}$$

Based on the above, generalized Dunn's index for a partition \mathbf{P} is defined as

$$Dunn_G(\mathbf{P}) = \frac{\min_{i,j,t \neq j} d_a(C_i, C_j)}{\max_h S_a(C_h)} \tag{4}$$

where $C_h, C_i, C_j \in \mathbf{P}$. The larger $Dunn_G(\mathbf{P})$ is, the better the partition (coloring).

2.3 b-Coloring Based Clustering

The notion of graph b-coloring was proposed in [12]. A b-coloring of an undirected graph G is a vertex coloring of G and satisfies the following two constraints:

- (i) adjacent vertices have different colors (proper coloring)
- (ii) for each color, there exists at least one vertex (called a **b-dominating vertex**) which is adjacent to all the other colors.

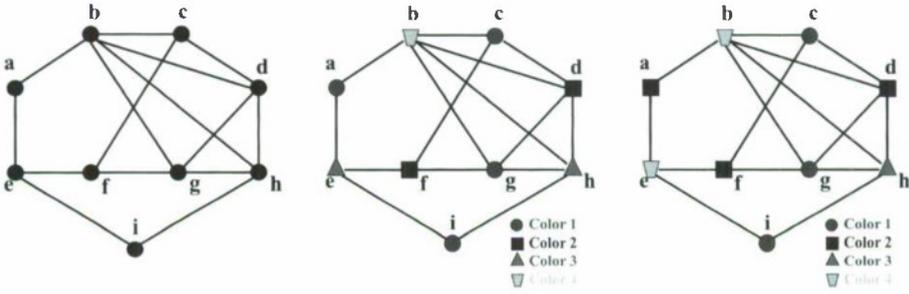


Fig. 1. A graph (with $\theta = 0.15$) for the data in Table 2 Fig. 2. A b-coloring of the graph in Fig. 1 ($Dunn_G = 0.916$) Fig. 3. Another b-coloring for Fig. 1 ($Dunn_G = 1.000$)

By assuming that some dissimilarity measure and a threshold θ are given, [5] proposed a clustering algorithm based on b-coloring. Each data item is mapped to a vertex, and vertices are connected if their dissimilarity is greater than θ . Thus, the entire data items are represented as a simple graph $G(V, E)$. For example, for the data items with dissimilarities in Table 2, Fig. 1 is the corresponding graph when θ is set to 0.15. Fig. 2 is an example of b-coloring of the graph in Fig. 1. This coloring is obtained by the algorithm in [5]. The vertices with the same color (shape) are grouped into the same cluster. Thus, $\{a,c,g,i\}$, $\{d,f\}$, $\{e,h\}$, $\{b\}$, are the clusters in Fig. 2.

Note that the graph is constructed such that the pairs of vertices with dissimilarity greater than θ are connected. Thus, adjacent vertices should be assigned to different clusters (colors), since they are “far away” from each other. This is guaranteed by the constraint (i). As the result, the data items within the same cluster are not dissimilar with each other. This corresponds to sustaining *intra-cluster cohesion*.

On the other hand, from (ii), each cluster contains at least one b-dominating vertex, which is adjacent to all the other clusters and thus is far (dissimilar) from them. This corresponds to sustaining *inter-cluster separation*. Especially, a b-dominating vertex in (ii) justifies the creation of the cluster with the vertex; since it cannot be assigned to all the other clusters, the cluster needs to be created to include it.

As explained in Section 2.2, finding out a partition with large *intra-cluster cohesion* and *inter-cluster separation* is important in clustering. These can be pursued in b-coloring based clustering via the constraints: the former by (i), and the latter by (ii).

2.4 Previous Re-coloring Method

Even for the same graph and the same number of clusters (colors), the graph in Fig. 1 has other different partition (b-coloring) with better quality (cf. with larger $Dunn_G$).

Table 2. An example of data set with dissimilarities

v_i	a	b	c	d	e	f	g	h	i
a	0								
b	0.20	0							
c	0.10	0.25	0						
d	0.15	0.2	0.25	0					
e	0.25	0.075	0.15	0.15	0				
f	0.15	0.10	0.2	0.15	0.25	0			
g	0.05	0.2	0.10	0.2	0.05	0.35	0		
h	0.05	0.20	0.15	0.30	0.15	0.10	0.2	0	
i	0.1	0.1	0.1	0.1	0.2	0.1	0.1	0.2	0

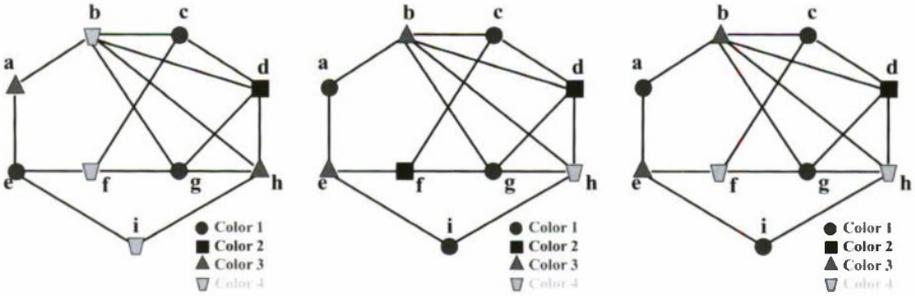


Fig. 4. Result with algorithm **Fig. 5.** Result with algorithm **Fig. 6.** Result with both algo-
 BFRColoring ($Dunn_G = ExColors (Dunn_G = 1.000)$ rithms ($Dunn_G = 1.750)$
 1.500)

An example of another b-coloring is shown in Fig. 3. The coloring in Fig. 3 ($Dunn_G = 1.000$) is better than that in Fig. 2 ($Dunn_G = 0.916$) w.r.t. the index in eq.(4).

In order to find better partitions, [6] proposed a greedy re-coloring algorithm. For a graph and its coloring, the colors of vertices are changed (re-colored) sequentially under the constraint that the number of b-dominating vertices is not decreased. For instance, for the graph and its coloring in Fig. 2, the greedy algorithm in [6] gives the b-coloring in Fig. 3 with better quality.

Effectiveness of the re-coloring based approach for obtaining better clusters was demonstrated in [6], however, the algorithm still has several limitations:

- i) greedy procedure: the colors of re-colored vertices were never modified again. Thus, other possibly better partitions could not be obtained.
- ii) vertices for re-coloring: only small portion of vertices were re-colored in order to guarantee the termination of the algorithm.
- iii) inaccurate quality estimation of clusters: not all vertices were utilized for quality estimation.

To cope with these issues, we propose an extended re-coloring approach. As for i), we propose a best first re-coloring algorithm (Section 3.2) to realize the non-greedy search for a better partition. As for ii), we propose a color exchange algorithm (Section 3.3) so that more vertices can be tested for re-coloring. As for iii), instead of the subset of vertices, we utilize all the vertices in a graph for estimating the quality of the partition.

For instance, for the same graph and its coloring in Fig. 2, in the proposed approach, the colorings in Fig. 4 ($Dunn_G = 1.500$) and Fig. 5 ($Dunn_G = 1.000$) are obtained by the algorithms **BFRColoring** (Section 3.2) and **ExColors** (Section 3.3), respectively. These are better or at least with the same quality compared with Fig. 2 ($Dunn_G = 0.916$) and Fig. 3 ($Dunn_G = 1.000$). Furthermore, since these algorithms are orthogonal with respect to the re-colored vertices, by utilizing both of them, even a better partition can be obtained, as shown in Fig. 6 ($Dunn_G=1.750$).

3 Re-coloring Algorithms Based on Graph b-Coloring

3.1 Definitions

For a b-coloring of a graph $G(V, E)$, a set of vertices V_d consists of b-dominating vertices in the coloring. For each b-dominating vertex $v_d \in V_d$, if $v_s \in N(v_d)$ ¹ is the only vertex with the color $c(v_s)$ in $N(v_d)$, this vertex is called a **supporting vertex** of v_d . V_s denotes the set of supporting vertices in the coloring. The set of vertices $V_c = V_d \cup V_s$ are called **critical vertices**. On the other hand, the set of vertices $V_{nc} = V \setminus V_c$ are called **non-critical vertices**². These are summarized in Table 3.

Table 3. Notations for vertices

symbol	description
V_d	b-dominating vertex set
V_s	supporting vertex set
V_c	critical vertex set
V_{nc}	non-critical vertex set

The proposed algorithms re-color the vertices when the quality of clusters is improved. Note that it is not assumed which quality measure is utilized. In the following description, $q(\cdot)$ stands for a quality measure of a partition (cf. *Dunn_C*).

3.2 A Best First Re-coloring Algorithm

To realize non-greedy search, we utilize the best first search strategy, which has been widely utilized in AI communities, and select the best coloring among the candidate colorings of a graph $G(V, E)$. From the graph, a pre-defined number of vertices are selected according to the descending order of $d_a(v, c(v))$ in eq.(1) where $v \in V$. Here, $d_a(v, c(v))$ can be interpreted as to what extent the vertex v is an "outlier" for the currently assigned cluster $c(v)$. Thus, we select the vertex with the largest $d_a(v, c(v))$ so that it can be moved into the other cluster via re-coloring. The color of the selected vertex is re-colored in order to increase the quality of partition as long as the constraints in b-coloring are satisfied. The above processes are repeated for the specified number of iterations.

Only critical vertices were utilized for estimating the quality of the partition in [6]. However, this can result in unreliable quality estimation and misguide search directions. To alleviate this problem, we utilize all the vertices for quality estimation so that the algorithm works as an any-time algorithm and that only better partitions are returned. Note that when the color of a non-critical vertex is re-colored, some critical vertices can become non-critical, and vice versa. Thus, after re-coloring of a vertex, the status of vertices is checked and reflected in the following re-coloring process.

The proposed algorithm BFRColoring is summarized in Algorithm 1. In Algorithm 1, b stands for the branching number, l for the number of iterations. One re-colored partition is obtained at line 6 by calling ReColoring in Algorithm 2. In Algorithm 2, the vertex with the largest average dissimilarity is selected at line 4. The color with the best quality is assigned for the vertex at line 7 via ReColoring in Algorithm 2.

¹ $N(v_d)$ returns the set of adjacent vertices to the vertex v_d .

² \setminus denotes set difference.

Algorithm 1. BFRColoring**Require:** $G(V, E)$ **Require:** P // a b -coloring partition of $G(V, E)$ **Require:** b // the branching number**Require:** l // the number of iterations

```

1:  $\mathcal{P}_{searched} \leftarrow \emptyset; \mathcal{P}_{cand} \leftarrow \emptyset; // \mathcal{P}$  represents a set of partitions
2:  $P_{current} \leftarrow P$ 
3: for  $i \leftarrow 0; i < l; i++$  do
4:    $\mathcal{P}_{searched} \leftarrow \mathcal{P}_{searched} \cup \{P_{current}\}$ 
5:   for  $j \leftarrow 0; (j < b); j++$  do
6:      $P^* \leftarrow reColoring(P_{current}, \mathcal{P}_{searched}) //$  call reColoring in Algorithm 2
7:      $\mathcal{P}_{cand} \leftarrow \mathcal{P}_{cand} \cup \{P^*\}$ 
8:   end for
9:    $\mathcal{P}_{searched} \leftarrow \mathcal{P}_{searched} \cup \mathcal{P}_{cand}$ 
10:   $P_{current} \leftarrow \arg \max_{P' \in \mathcal{P}_{cand}} q(P') //$   $q(\cdot)$  evaluates the quality of a partition
11:   $\mathcal{P}_{cand} \leftarrow \mathcal{P}_{cand} \setminus \{P_{current}\}$ 
12: end for
13: return  $\arg \max_{P' \in \mathcal{P}_{searched}} q(P')$ 

```

For a graph, computation of $d_a(v_i, c(v_i))$ for all the vertices can be conducted in $O(n^2)$ at the beginning and it can be updated in $O(n)$ at line 4 in Algorithm 2. By denoting the time complexity of quality evaluation $q(\cdot)$ as p^3 , line 7 takes at most $O(\Delta p)$ since $|C_p(v^*)| \leq \Delta + 1$. Algorithm 2 can take $O(n(n + \Delta^2 p))$ in the worst case when both while loops at lines 3 and 6 are exhaustively iterated. However, this is rather too pessimistic estimation, since these while loops are for avoiding the duplicated partitions. Thus, in most cases the most expensive process (line 7) is called only once in Algorithm 2. Thus, complexity of Algorithm 1 can be considered as $O(bl\Delta p)^4$.

3.3 A Color Exchange Algorithm

If the color of a critical vertex is changed, the number of b -dominating vertices will decrease. Since b -dominating vertices are considered as useful for sustaining inter-cluster separation, re-coloring was conducted only on non-critical vertices in [6].

Although it is difficult to re-color critical vertices *sequentially* without decreasing the number of b -dominating vertices, this problem can be resolved if more than one vertices are re-colored *simultaneously*. As a first step, we propose a color exchange algorithm for critical vertices. We define that two adjacent critical vertices are color exchangeable if the following three conditions are satisfied.

Definition 1 (Color Exchangeable). For a graph and its partition (coloring) P , let P' be the coloring by exchanging the colors of two adjacent critical vertices. If P' satisfies the followings, these vertices are called color exchangeable:

³ $Dunn_G$ can be calculated in $O(n^2)$ and updated in $O(n)$ for re-coloring of a vertex.

⁴ Admittedly, $O(bl(n(n + \Delta^2 p)))$ in the worst case in standard notation.

Algorithm 2. reColoring

Require: $P_{current}$ // a b-coloring partition of $G(V, E)$
Require: $\mathcal{P}_{searched}$ // a set of searched partitions

- 1: $P' \leftarrow P_{current}$ // copy the coloring (partition)
- 2: $V' \leftarrow V_{nc}$ // candidate vertices in P'
- 3: **while** $V' \neq \emptyset$ **do**
- 4: $v^* := \arg \max_{v \in V'} d_a(v, c(v))$
- 5: $C' \leftarrow \emptyset$ // C' stores the tested colors of v^*
- 6: **while** $C_p(v^*) \setminus C' \neq \emptyset$ **do**
- 7: $c^*(v^*) \leftarrow \arg \max_{c \in C_p(v^*) \setminus C'} q(P(v^*, c))$ // $P(v^*, c)$ is a partition with color c for v^*
- 8: $C' \leftarrow C' \cup \{c^*(v^*)\}$ // add to the already tested colors
- 9: re-color $c(v^*)$ to $c^*(v^*)$ in P'
- 10: **if** $P' \notin \mathcal{P}_{searched}$ **then**
- 11: **return** P' // return the re-colored partition
- 12: **end if**
- 13: re-color $c^*(v^*)$ back to the original $c(v^*)$ in P' // P' was already searched
- 14: **end while**
- 15: $V' \leftarrow V' \setminus \{v^*\}$
- 16: **end while**
- 17: **return** \emptyset

1. all the adjacent vertices have different colors,
2. the number of b-dominating vertices is not decreased,
3. the number of colors is not decreased.

Currently, candidate vertices for color exchange are: 1) a b-dominating vertex v_i and its supporting vertex v_j , or, 2) for some b-dominating vertex, its two supporting vertices v_j and v_k . For a b-dominating vertex v_i , a supporting vertex v_j is the only vertex with color $c(v_j)$ in $N(v_i)$. Thus, if the color $c(v_i)$ is different from the neighboring colors of v_j , exchanging their colors does not decrease the number of b-dominating vertices. Similarly, for a b-dominating vertex, exchanging the colors of its two supporting vertices does not decrease the number of b-dominating vertices.

The proposed algorithm ExColors is summarized in Algorithm 3. For the selected vertex, at line 5 the candidate vertices for color exchange are enumerated using ExVertices in Algorithm 4. Color exchange is conducted if a) the pair of critical vertices are color exchangeable, and b) the quality of partition would be improved (line 9). If there are more than one vertex for exchange, the vertex with the maximum average dissimilarity is selected.

As in Algorithm 2, selection of a vertex can be conducted in $O(n)$ at line 3 in Algorithm 3. Since up to two-step neighboring vertices for the selected vertex are checked in Algorithm 4, at most $O(\Delta^2)$ vertices are obtained as the candidates. Thus, the overall time complexity of Algorithm 3 is $O(n(n + \Delta^2 p))^5$.

⁵ As in Section 3.2, time complexity of $q(\cdot)$ is denoted as p .

Algorithm 3. ExColors**Require:** $G(V, E)$ //A graph which a set of vertices and a set of edges**Require:** P //a partition which is a b-coloring of $G(V, E)$

```

1:  $V' \leftarrow V_c$ 
2: while  $V' \neq \emptyset$  do
3:    $v_i := \arg \max_{v_i \in V'} d_a(v_i, c(v_i))$ 
4:    $V' \leftarrow V' \setminus \{v_i\}$ 
5:    $V'_{ex} := \text{ExVertices}(G, P, v_i)$  // call ExVertices in Algorithm 4
6:   while  $V'_{ex} \neq \emptyset$  do
7:      $v_j := \arg \max_{v_j \in V'_{ex}} d_a(v_j, c(v_j))$ 
8:      $V'_{ex} \leftarrow V'_{ex} \setminus \{v_j\}$ 
9:     if  $(c(v_i) \text{ and } c(v_j) \text{ is exchangeable}) \wedge (q(P) < q(P(\text{exchange}(c(v_i), c(v_j))))$  then
10:      exchange color  $c(v_i)$  and  $c(v_j)$  in  $P$ 
      // the colors of  $v_i$  and  $v_j$  are exchangeable and the quality would be improved
11:       $V' \leftarrow V_c$ ; break;
12:     end if
13:   end while
14: end while
15: return  $P$ 

```

3.4 Working Examples

As shown in Section 2.4, for the graph and its coloring in Fig. 2, the colorings in Fig. 4 ($Dunn_G = 1.500$) and Fig. 5 ($Dunn_G = 1.000$) are obtained by BFRColoring and ExColors, respectively. Non-critical vertices a, e, f, i were re-colored in Fig. 4. The colors of critical vertices b and h were exchanged in Fig. 5.

Furthermore, critical vertices are considered for color exchange in ExColors; on the other hand, non-critical vertices are re-colored in BFRColoring. Since $V_c \cap V_{nc} = \emptyset$, these are mutually independent and orthogonal with respect to the re-colored vertices. Thus, these can be utilized in conjunction. The coloring in Fig. 6 (with $Dunn_G=1.750$) is obtained by applying both algorithms. In this example, critical vertices b and h, a non-critical vertex f were re-colored. Thus, the proposed approach enables to obtain better partitions (colorings) by enlarging the search space via non-greedy search and re-coloring of critical vertices.

4 Preliminary Evaluations

4.1 Evaluation Measures

In addition to $Dunn_G$ in eq.(4), we also evaluated a) micro-averaged Precision, and b) distinctness, of a partition. As in $Dunn_G$, the larger the evaluated value is, the better the partition is.

Micro-Averaged Precision. Micro-averaged precision is a widely utilized measure in information retrieval community [1]. Based on the cross table of true clusters and

Algorithm 4. ExVertices

Require: $G(V, E)$ //A graph which a set of vertices and a set of edges

Require: P //a partition which is a b-coloring of $G(V, E)$

Require: v_i : a vertex

```

1: if  $v_i \in V_d$  then
2:    $V'_{ex} :=$  supporting vertices of  $v_i$ 
3: else if  $v_i \in V_s$  then
4:   for each  $v_n \in N(v_i)$  do
5:     if  $(v_n \in V_d) \wedge (v_i$  is a supporting vertex of  $v_n)$  then
6:        $V'_{ex} := \{v_n\} \cup$  supporting vertices of  $v_n$ 
7:     end if
8:   end for
9: end if
10: return  $V'_{ex}$ 

```

assigned clusters, it is calculated by averaging the precision of data assignment to each constructed cluster. Please refer to [1] for the details. We call this Precision hereafter.

Distinctness. The variance of the distribution match between clusters C_h and C_l in a partition is defined as:

$$Var(C_h, C_l) = \frac{1}{p} \sum_i^p \sum_j^p (P(a_i = x_{ij}|C_h) - P(a_i = x_{ij}|C_l))^2 \tag{5}$$

where p is the number of attributes. $P(a_i = x_{ij}|C_l)$ represents the conditional probability of attribute a_i taking the value x_{ij} in cluster C_l .

The distinctness of a partition P is defined as the average variance [14]:

$$Dist(P) = \frac{\sum_{h=1}^k \sum_{l=1}^k Var(C_h, C_l)}{|P|(|P| - 1)} \tag{6}$$

4.2 Experimental Settings

Preliminary evaluations were conducted over several UCI datasets [11]. The utilized datasets were: Zoo (101 data, 7 labels), Teaching Assistant Evaluation (tae) (151 data, 3 labels), and Protein Localization Sites (ecoli) (336 data, 8 labels). In all the datasets, each data item has its true class label. The true class labels are regarded as “ground truth” and utilized to calculate Precision. After normalizing each attribute to [0,1] as in Weka [17]), dissimilarities between data items were calculated using the standard Euclidian distance.

The proposed algorithms (with BFRColoring, with ExColors, with both of them) were compared with the following clustering algorithms: 1) previous re-coloring algorithm [6], 2) kmeans algorithm [10], and 3) EM algorithm [3]. Weka [17]) was used for kmeans and EM. Since kmeans and EM require the number of clusters, the true number of clusters was specified for each dataset.

Table 4. Result (zoo dataset)

	<i>Prec</i>	<i>Dist</i>	<i>Dunn_g</i>
BF	0.812	0.479	1.050
Ex	0.743	0.577	0.796
Ex+BF	0.812	0.479	1.050
greedy	0.733	0.401	0.910
kmeans	0.723	0.489	1.014
EM	0.673	0.591	0.981

Table 5. Result (tae dataset)

	<i>Prec</i>	<i>Dist</i>	<i>Dunn_g</i>
BF	0.444	0.363	0.983
Ex	0.430	0.355	0.923
Ex+BF	0.444	0.363	0.983
greedy	0.430	0.530	1.350
kmeans	0.517	0.458	1.132
EM	0.404	0.321	1.141

Table 6. Result (ecoli dataset)

	<i>Prec</i>	<i>Dist</i>	<i>Dunn_g</i>
BF	0.631	0.444	0.831
Ex	0.591	0.449	0.653
Ex+BF	0.631	0.444	0.831
greedy	0.324	0.419	1.004
kmeans	0.613	0.153	0.609
EM	0.619	0.168	0.604

Following the experimental setting in [6], the same graph and its partition (coloring of the graph) were given to the proposed algorithms and 1), and $Dunn_G(\cdot)$ was used as the quality measure $q(\cdot)$ in the algorithms. In Algorithm 1, b was set to 10 and l was set to 10^3 . The threshold for defining the graph structure was set so that the same number of colors (clusters) was obtained in each dataset.

4.3 Results

The results are summarized in Tables 4, 5, 6. In the tables, **BF** stands for BFRColoring (Algorithm 1), **Ex** stands for ExColors, (Algorithm 3), **Ex+BF** stands for applying ExColors and BFRColoring in this order, **greedy** stands for the algorithm in [6].

The results show that the proposed algorithms outperform the other algorithms in most cases w.r.t. Precision. Since the evaluation based on the true class label is considered as the so-called “ground truth” evaluation, the results indicate that the proposed approach is promising toward improving re-coloring based clustering.

Intuitively, $Dist(\cdot)$ in eq.(6) evaluates to what extent the obtained clusters differ w.r.t. the prediction of the attribute value. The results varied depending on the datasets and it is difficult to draw a decisive conclusion w.r.t. distinctness from the results.

$Dunn_G(\cdot)$ in eq.(4) was used as the quality measure $q(\cdot)$ in our algorithms and **greedy**. These algorithms improved this quality, and the values were larger than those obtained by **kmeans** and **EM** (except for tae dataset). However, in Table 6, **greedy** returned the largest $Dunn_G(\cdot)$ value, but it is actually the worst w.r.t. Precision.

4.4 Discussion

Results in Section 4.3 indicate that the proposed approach is effective for improving the performance of re-coloring based clustering in terms of Precision. Unfortunately, Precision cannot be utilized as the quality measure $q(\cdot)$ in any algorithms *directly*, since it is calculated based on the “true” labels. Note that “true” labels are unavailable for clustering or in unsupervised learning in general. On the other hand, eq.(4) can be calculated only from the available data and thus can be utilized. It is not yet clear how the latter correlates with Precision. In addition, the performance of **BF** and **Ex+BF** was the same for these datasets. Much more work needs to be conducted for investigating the usage of dissimilarity information in the algorithm.

5 Conclusion

This paper has proposed an approach toward improving re-coloring based clustering with graph b-coloring. Based on the notion of b-coloring in graph theory [12], clustering algorithms were proposed in previous approach; however, these were still restrictive in terms of the explored search space due to its greedy and sequential re-coloring process. In this paper a best first re-coloring algorithm was proposed to realize non-greedy search for the admissible colors of vertices. A color exchange algorithm was proposed to remedy the problem in sequential re-coloring. Both algorithms enlarge the search space and re-color the vertices of a graph to improve the quality of clusters, while guaranteeing the property of b-coloring. In addition, these algorithms are orthogonal with respect to the re-colored vertices and thus can be utilized in conjunction.

Preliminary evaluations were conducted over several UCI datasets. The results are encouraging for pursuing this line of research, especially for obtaining better clusters with respect to the ground truth micro-averaged precision. However, with respect to other clustering validation indices, it was rather comparable to other approaches and could not always outperform them. We plan to conduct more evaluations and investigate the suitable quality measure to guide the search process in the proposed algorithms.

Acknowledgments

This work is partially supported by the grant-in-aid for scientific research (No. 20500123) funded by MEXT, Japan.

References

1. Baeza, Y., Ribeiro, N.: *Modern Information Retrieval* (1999)
2. Bezdek, J., Pal, N.: Some new indexes of cluster validity. *IEEE Transactions on Systems, Man and Cybernetics* 28(3), 301–315 (1998)
3. Dempster, A., Laird, N., Rubin, D.: Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society* 39, 1–38 (1977)
4. Diestel, R.: *Graph Theory*. Springer, Heidelberg (2006)
5. Elghazel, H., Deslandres, V., Hacid, M., Dussauchoy, A., Kheddouci, H.: A new clustering approach for symbolic data and its validation: Application to the healthcare data. In: Esposito, F., Raś, Z.W., Malerba, D., Semeraro, G. (eds.) *ISMIS 2006. LNCS (LNAI)*, vol. 4203, pp. 473–482. Springer, Heidelberg (2006)
6. Elghazel, H., Yoshida, T., Deslandres, V., Hacid, M., Dussauchoy, A.: A new greedy algorithm for improving b-coloring clustering. In: *Proc. of the GbR 2007*, pp. 228–239 (2007)
7. Guénoche, A., Hansen, P., Jaumard, B.: Efficient algorithms for divisive hierarchical clustering with the diameter criterion. *Journal of Classification* 8, 5–30 (1991)
8. Guha, S., Rastogi, R., Shim, K.: Cure: An efficient clustering algorithm for large databases. In: *Proceedings of the ACM SIGMOD Conference*, pp. 73–84 (1998)
9. Hansen, P., Delattre, M.: Complete-link cluster analysis by graph coloring. *Journal of the American Statistical Association* 73, 397–403 (1978)
10. Hartigan, J., Wong, M.: Algorithm as136: A k-means clustering algorithm. *Journal of Applied Statistics* 28, 100–108 (1979)

11. Hettich, S., Blake, C., Merz, C.: UCI repository of machine learning databases (1998)
12. Irving, W., Manlov, D.F.: The b-chromatic number of a graph. *Discrete Applied Mathematics* 91, 127–141 (1999)
13. Jain, A.K., Murty, M.N., Flynn, P.J.: Data clustering: A review. *ACM Computing Surveys* 31, 264–323 (1999)
14. Kalyani, M., Sushmita, M.: Clustering and its validation in a symbolic framework. *Pattern Recognition Letters* 24(14), 2367–2376 (2003)
15. Ng, R., Han, J.: Clarans: a method for clustering objects for spatial data mining. *IEEE Transactions on Knowledge and Data Engineering* 14(5), 1003–1016 (2002)
16. von Luxburg, U.: A tutorial on spectral clustering. *Statistics and Computing* 17(4), 395–416 (2007)
17. Witten, I., Frank, E.: Weka, <http://www.cs.waikato.ac.nz/ml/weka/>

Semi-supervised Constrained Clustering: An Expert-Guided Data Analysis Methodology

Vid Podpečan¹, Miha Grčar¹, and Nada Lavrač^{1,2}

¹ Jožef Stefan Institute, Ljubljana, Slovenia

² University of Nova Gorica, Nova Gorica, Slovenia

Abstract. This paper presents a methodology for expert-guided analysis of large data sets, including large text corpora. Its main ingredient is the algorithm for semi-supervised data clustering using cluster size constraints which implements several improvements over existing k-means constrained clustering algorithms. First, it allows for a larger set of user-defined cluster size constraints of different types (lower- and upper-bound constraints). Second, it allows for dynamic re-assignment of predefined constraints to clusters in iterative cluster computation/optimization, thus improving the results of constrained clustering. Third, it allows for expert-guided cluster optimization achieved by combining constrained clustering and data visualization, which enables finer-grained expert's control over the clustering process, leading to further improvements of the quality of obtained clustering solutions. Incorporating data visualization into the clustering process allows the user to select referential points which act as constraint anchors in the course of iterative cluster computation. The proposed semi-supervised constrained clustering methodology has been implemented using a service-oriented data mining environment Orange4WS and evaluated on different document corpora.

1 Introduction

Clustering is a method of unsupervised learning, aimed at assigning a set of data instances into subsets called clusters so that instances in the same cluster are similar according to a predefined similarity measure. K-means clustering [8] has proven to be an effective tool both in data and text mining. In text mining, k-means clustering is being used extensively for exploratory text analysis including concept identification [9] and document corpora visualization [14]. Although widely used because of its speed and simplicity, the k-means clustering algorithm and its variants have some serious drawbacks which limit their use in specific scenarios.

The most popular version of k-means, i.e. the Forgy's algorithm [1,8] is known to produce unbalanced and/or empty clusters when applied to datasets with a high number of dimensions and a large number of clusters [3,8]. For example, clustering of Web browsing data [3] with 300 dimensions (features) resulted on average in 4.1 and 12.1 empty clusters where k was set to 50 and 100, respectively. More generally, this phenomenon was observed when clustering data with the

number of dimensions $n \geq 10$, where the number of desired clusters was set to $k \geq 20$ [3]. The problem of empty and unbalanced clusters has been addressed in the area of constrained clustering, briefly introduced below.

1.1 Constrained Clustering

Constrained clustering is a class of semi-supervised learning algorithms which can be divided into two main groups. Clustering algorithms with instance-based constraints typically incorporate a set of must-link constraints and/or cannot-link constraints [19]. Clustering algorithms with cluster-based constraints [3,18], on the other hand, incorporate constraints concerning the size or shape of individual clusters. In order to address the problem of empty clusters mentioned above, Bradley, Bennett, and Demiriz [3] proposed a constrained clustering algorithm, explicitly adding k constraints to the underlying optimization problem which state that each cluster h should contain at least τ_h points. By integrating these constraints into the optimization procedure, they present a clear, mathematically well-formed solution which can be also generalized to other constraints (e.g. outlier removal or specific groupings).

In this paper, we present a method for semi-supervised constrained data clustering using cluster size constraints, upgrading the k -means clustering method. To do so, we first briefly present the k -means algorithm with additional constraints. For the sake of clarity, the same notation as in [3] is used throughout this introductory section.

Let $\mathcal{D} = \{x^i, i = 1, \dots, m\}$ be a dataset in \mathbb{R}^n and k the desired number of clusters. Then, the problem of k -means clustering is to find cluster centers C^1, C^2, \dots, C^k where the sum of the squared error¹ (SSE) is minimized [17]. More formally, this can be written as:

$$\min_{C^1, \dots, C^k} \sum_{i=1}^m \min_{h=1, \dots, k} \text{dist}(x^i, C^h) \quad (1)$$

This equation, however, can be reformulated into an equivalent form where binary selector variables $T_{i,h}$ are introduced. These variables indicate the membership of data points to clusters: $T_{i,h} = 1$ if data point x^i is closest to center C^h and zero otherwise. The reformulated Eq. 1 is then as follows [3]:

$$\begin{aligned} \underset{C, \mathcal{T}}{\text{minimize}} \quad & \sum_{i=1}^m \sum_{h=1}^k T_{i,h} \cdot \text{dist}(x^i, C^h) \\ \text{where} \quad & \sum_{h=1}^k T_{i,h} = 1, i = 1, \dots, m \\ & T_{i,h} \geq 0, i = 1, \dots, m; h = 1, \dots, k \end{aligned} \quad (2)$$

The proof that the new equation with selector variables is equivalent to the original can be found in [4] as Lemma 2.1. Note that it is possible for the k -means algorithm to produce one or more empty clusters i.e. $\sum_{i=1}^m T_{i,h} = 0$ as such a solution satisfies the Karush-Kuhn-Tucker (KKT) conditions [17] for Eq.

¹ SSE is also known as *scatter*.

2. The constrained k-means algorithm can be now formalized by adding the following cluster size constraints to Eq. 2:

$$\sum_{i=1}^m T_{i,h} \geq \tau_h \quad (3)$$

Values represented by τ_h are constants specified in advance by the user. In plain terms, each cluster h must contain at least τ_h data points. To assure that the constructed optimization problem is solvable we add a sanity condition $\sum_{h=1}^k \tau_h \leq m$ which states that the sum of all size constraints is not larger than the size of the observed set of data instances.

Finally, the constrained k-means algorithm is, like the classic k-means, defined as an iterative two-step procedure which iterates between solving the linear program defined by Eq. 2 and 3 to obtain values for selector variables $T_{i,h}$ (cluster assignment step), followed by updating the cluster centers C^h (cluster update step). As a last remark on the constrained k-means algorithm the following statements were proven to be true [3]:

1. The constrained k-means algorithm terminates in a finite number of iteration in a locally optimal cluster assignment.
2. The cluster assignment sub-problem (step 1 of each iteration) is equivalent to the Minimum Cost Flow (MCF) network optimization problem.
3. According to statement (2) above and [2] the optimal flow of the equivalent MCF problem is integer-valued which means that the optimal binary values for $T_{i,h}$ can be obtained without explicitly declaring them as integer thus solving the integer programming problem (ILP) which is know to belong to the $\mathcal{NP} - hard$ class of problems [11].

1.2 Summary of Research Advances and Paper Outline

The main contribution of this paper is a methodology for expert-guided analysis of large data sets, including large text corpora. Its main ingredient is the algorithm for semi-supervised data clustering using cluster size constraints which successfully eliminates some limitations of existing k-means constrained clustering algorithms. First, it allows for a larger set of user-defined cluster size constraints of different types (lower- and upper-bound constraints). Second, it allows for dynamic re-assignment of predefined constraints to clusters in iterative cluster computation/optimization. Third, it allows for expert-guided cluster optimization. The proposed semi-supervised constrained clustering algorithm is presented in Section 2. Expert-guidance is achieved by combining constrained clustering and data visualization, which enables finer-grained expert's control over the clustering process, leading to further improvements of the quality of obtained clustering solutions. Incorporating data visualization into the clustering process, as described in Section 3, allows the user to explore the data and to select referential points representing initial cluster centroids, which (in the simplest scenario) act also as constraint anchors in the course of iterative cluster

computation. The proposed semi-supervised constrained clustering methodology has been implemented using a service-oriented data mining environment Orange4WS [15]. The evaluation of the methodology on various text corpora is presented in Section 4. The paper concludes with a summary and plans for further work.

2 Semi-supervised k-Means with Cluster Size Constraints

The constrained variant of the k-means algorithm, presented in Section 1, is not appropriate if a certain cluster size constraint needs to be assigned to a cluster with specific semantics (e.g. cluster containing documents discussing a certain topic). During the clustering process, cluster centers “travel” in the direction, opposite to the gradient of the target function in the observed space which means that each specified constraint τ_h will apply to an unknown part of the space with input data. The only applicable scenario (which was also addressed by the authors [3]) is the case with balanced constraints where all τ_h are equal. (in this special case one is not concerned with the size of an individual cluster, the objective is just to eliminate empty or very small clusters).

Therefore, we propose a modified algorithm (Algorithm 1), which is able to overcome the indicated problem. To this end, our variant of the algorithm maintains points of reference with respect to the given constraints and modifies the optimization problem specifications accordingly. It should be noted, however, that the new variant requires certain amount of domain knowledge (user’s background knowledge) in order to be applied successfully. In the context of clustering document corpora, which is the target domain of this paper, such knowledge can be provided by visualization, as shown in Section 3 below.

The idea of the proposed algorithm is the following. In order to apply constraints to specific parts of the data space, there have to exist the same number of reference points, one for each constraint. Each such reference point characterizes the part of the input data space where the constraint should be enforced. However, as cluster centroids tend to travel through the data space during the clustering process, the constraints are likely to be applied to a completely different part of the space than the initial data subspace. For this reason, our algorithm recomputes distances between reference points and the current centroids in each iteration and reassigns constraints when necessary. The proposed modification of the constrained k-means algorithm is presented as Algorithm 1.

Note, however, that reassignments of constraints modifies the underlying optimization problem which introduces the possibility of cycling where clusters exchange constraints without converging their centroids to final positions (the Proposition 1 from Section 1 stating that the algorithm finishes in a finite number of steps no longer holds). Although such situations are very unlikely to occur in high dimensional data spaces with good initial centroids, a solution in these rare cases is to employ simulated annealing with a simple cooling schedule or to introduce small random jitter of centroids. As already stated, a set of reference points \mathbf{P} , representing both initial cluster centroids and reference points for con-

Algorithm 1

Input:

- data set in \mathbb{R}^n with m instances: $\mathcal{D} = \{x^i, i = 1, \dots, m\}$
- desired number of clusters: k
- set of constraints: $\tau = \{\tau_1, \dots, \tau_k\}$
- set of reference points in \mathbb{R}^n : $\mathcal{P} = \{p^i, i = 1, \dots, k\}$ which is also the set of initial centroids: $\mathcal{C}^0 = \{C^{i,0}, i = 1, \dots, k\}$

Output of iteration t :

- assignment of input data instances to clusters with respect to given constraints
- set of centroids: $\mathcal{C}^t = \{C^{i,t}, i = 1, \dots, k\}$

Each iteration t of the algorithm consists of three steps:

1. Cluster assignment.

Solve a linear program to obtain the values of selector variables $T_{i,h}^t$:

$$\begin{aligned} & \underset{C, T}{\text{minimize}} && \sum_{i=1}^m \sum_{h=1}^k T_{i,h}^t \cdot \text{dist}(x^i, C^{h,t}) \\ & \text{where} && \sum_{h=1}^k T_{i,h}^t = 1, \quad i = 1, \dots, m \\ & && T_{i,h}^t \geq 0, \quad i = 1, \dots, m; \quad h = 1, \dots, k \\ & && \sum_{i=1}^m T_{i,h}^t \geq \tau_h \quad h = 1, \dots, k \end{aligned}$$

2. Cluster update.

Compute new centroids for the next iteration $t+1$:

$$C^{h,t+1} = \begin{cases} \frac{\sum_{i=1}^m T_{i,h}^t x^i}{\sum_{i=1}^m T_{i,h}^t} & \text{if } \sum_{i=1}^m T_{i,h}^t > 0 \\ C^{h,t} & \text{otherwise} \end{cases}$$

3. Permutation of constraints.

Assign newly computed centroids $C^{h,t+1}$ to reference points p^i by computing binary selector variables $V_{h,i}^{t+1}$ so that the total distance is minimized, and permute assignment of constraints to clusters accordingly:

$$\begin{aligned} & \text{minimize} && \sum_{i=1}^k \sum_{h=1}^k V_{h,i}^{t+1} \cdot \text{dist}(p^i, C^{j,t+1}) \\ & \tau_i = && \begin{cases} \tau_j & \text{if } V_{h,i}^{t+1} \text{ is true} \\ \tau_i & \text{otherwise} \end{cases} \end{aligned}$$

straints, is required as input to the algorithm. In order to specify these points, the user needs to have an understanding of the underlying data. To provide the user with a better understanding of the underlying data, we employ a feature space visualization algorithm based on least-squares meshes [16,14], described in Section 3 below. Through data visualization, the user is able to anchor the cluster size constraints to specific parts of interest in the data space.

3 Methodology for Expert-Guided Constrained Clustering Facilitated by Data Visualization

This section presents the proposed semi-supervised constrained clustering methodology. In addition improving the constrained clustering algorithm (Algorithm 1), the main additional assets used in this process are data visualization and user-guided constrained clustering through an interface to the visualized data clouds, enabling initial centroid selection and size constraints specification. This section first outlines the steps of the proposed methodology, followed by presenting the algorithm which enables the visualization of the data space, more specifically, a document space using the bag-of-words document representation. In the context of this paper, the process of visualization is seen as a procedure which extracts knowledge about the underlying structure of the data. The visualization method should namely be able to provide enough information to guide the expert when specifying the constraints and should also (implicitly) help the clustering algorithm to converge faster by providing good initial centroids. The document corpora visualization method presented in this section is a combination of multidimensional scaling, least-squares solver, and internal k-means clustering.

3.1 Methodology

The proposed semi-supervised constrained clustering methodology consists of the following main steps:

1. The input data is preprocessed as required by the clustering and visualization algorithms.
2. The least-squares meshes data visualization algorithm is invoked. As a result, the user is presented with a 2D projection of high-dimensional data instances, such as the one presented in Figure 1a.
3. The graphical user interface of our algorithms enables the user to visually identify centers of condensed groups of data instances and to anchor constraints to these points, called reference points (visualized as triangles in Figure 1b). Furthermore, the user defines each of the constraints by setting the lower- and/or upper-size limit of the corresponding cluster.
4. When the constraints are fully specified, the constrained clustering algorithm is invoked. The algorithm takes reference points (i.e. constraint anchors) as the initial centroid locations. The centroids then travel around the space during the optimization process, while the reference points (and thus the constraints) keep their initial positions. In each step of the clustering process, the constraints can be reassigned to centroids (if necessary) according to the constraint permutation step in Algorithm 1.
5. The algorithm outputs the size-constrained data clusters to be further inspected (and possibly refined) by the user.

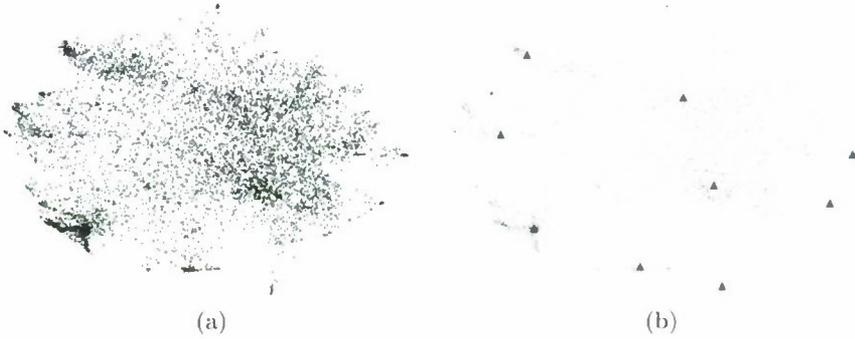


Fig. 1. 2D projection of high-dimensional data instances of the Yahoo Finance dataset (a) with manually selected reference points (b)

3.2 Visualization of Large Document Corpora

A bag-of-words document space is a high-dimensional space in which documents are represented as feature vectors (TF-IDF vectors). To visualize the bag-of-words space, we need to project feature vectors onto a 2-dimensional canvas so that the distances between the planar points reflect the cosine similarities between the corresponding feature vectors.

For the purpose of this visualization, we followed the work of Sorkine and Cohen-Or [16] and Panlovich et al. [14] which is based on least-squares meshes (for this reason, we use the term least-squares meshes visualization throughout this paper). To compute the projection of high-dimensional feature vectors onto a planar canvas, several methods are employed in a pipeline. Clustering of the feature vectors is first performed to obtain several smaller, more manageable segments of the feature space. Then, several representative instances – medoids of the obtained clusters – are selected and their layout is computed. As the number of representative instances r is much smaller than the number of feature vectors n ($r \ll n$), computationally expensive techniques can be employed for this purpose. In our case, stress majorization [10] is employed to perform this step of the process. After the representative instances are positioned in 2D, a system of linear equations is constructed and solved in the least-squares sense. The solution of the system represents the projection of all the feature vectors onto a planar canvas. To construct a system of linear equations, planar coordinates of several control points (obtained by the stress majorization algorithm) and the k nearest neighbors of each instance are required. In Eq. 4, P_i denotes a point (both coordinates) and N_{P_i} the set of its nearest neighbors (note that a point is not its own nearest neighbor).

$$P_i = \frac{1}{|N_{P_i}|} \sum_{S \in N_{P_i}} S \quad (4)$$

Instances of Eq. 4 for all points and precomputed positions of control points can be expressed as a system of sparse linear equations, as shown² in Eq. 5.

$$\begin{bmatrix} \mathbf{A} \\ \mathbf{A}' \end{bmatrix} \mathbf{X} = \begin{bmatrix} \mathbf{0} \\ \mathbf{B} \end{bmatrix} \quad (5)$$

Here, (sub)system $\mathbf{AX}=\mathbf{0}$ contains instances of Eq. 4 and (sub)system $\mathbf{A}'\mathbf{X}=\mathbf{B}$ defines (known) positions of control points. Vector \mathbf{X} represents unknown final positions of points, respectively. Note that the system, defined by Eq. 5, is overdetermined³. As such systems usually have no solution, the goal is to find a “solution” which fits the equations best in the least squares sense. In our document stream visualization framework the LSQR solver, developed by Paige and Saunders [13], was used to obtain the solution of Eq. 5 which is a set of planar points corresponding to the high-dimensional feature vectors.

3.3 Implementation

The proposed data analysis methodology was implemented using a combination of various technologies and open source software libraries. Firstly, the data preprocessing step was implemented in Python. Secondly, the Orange4WS web service environment [15] was employed to invoke web services, built on top of the LATINO multilingual text mining library⁴ which provides all the required components to produce sparse vector representation of textual data and their visualization: tokenizers, lemmatizers/stemmers, n-gram detection, bag-of-words computation, and the least-squares meshes visualization method. The clustering algorithm was implemented in Python using the *numpy* package⁵ for numerical computations and Python interface to the *lp_solve* mixed integer linear programming (MILP) solver⁶ which essentially forms the backbone of the constrained k-means clustering algorithm. The graphical user interface was written in Python using cross-platform open source framework Qt⁷ and its extension for technical applications named Qwt⁸.

4 Evaluation

The proposed methodology is illustrated through constrained clustering tasks using four datasets: the Yahoo finance dataset⁹ with 6177 short company descriptions, Inductive Logic Programming (ILP) dataset¹⁰ with 1407 scientific

² For the sake of clarity, this system combines all dimensions of points (vectors X and B have dimensions $[n \times 2]$). In practice, we have to solve such a system for each dimension separately.

³ Overdetermined systems of linear equations have more equations than variables.

⁴ <http://sourceforge.net/projects/latino>

⁵ <http://numpy.scipy.org>

⁶ <http://lpsolve.sourceforge.net/5.5/>

⁷ <http://qt.nokia.com/>

⁸ <http://qwt.sourceforge.net/>

⁹ Available at http://ontogen.ijs.si/?page_id=10

¹⁰ Available at <http://www.cs.bris.ac.uk/~ILPnet2/>

publications out of which 506 contain both titles and abstracts and the rest contain titles only, and a corpus containing the Proceedings of the Slovenian Informatics Conference¹¹ (DSI) from 2003 to 2009 with 833 texts in slovene language (the use of this dataset also demonstrates multilingual abilities of our implementation). Figure 2 shows the visualization of all four datasets using the least-squares meshes method. Clearly, least-squares meshes visualization provides enough information to identify potential clusters, their approximate sizes and initial centroids. This visual information was used to set up constraints for the constrained clustering scenario. For example, in the Yahoo Finance dataset, nine clusters were identified by the user and the corresponding size constraints were defined through visual assessment by using the graphical user interface. Figure 1b presents the positions of our reference points carrying cluster size constraints.



Fig. 2. Visualization of datasets: (a) Yahoo Finance (6,177 instances), (b) subset of ILP-Inductive Logic Programming with abstracts available (506 instances), (c) subset of ILP with only titles available (1401 instances) ILP (d) DSI-Slovenian Informatics Conference (833 instances)

Table 1 summarizes the experimental results. Our semi-supervised algorithm was compared to ordinary k-mean using the same initial centroids, and to ordinary k-means using random initial centroids¹². The number of clusters was determined visually (least-squares meshes visualization) by identifying dense components and well-separated parts of data space. We used the Davies-Bouldin cluster validity measure [5] which is a function of the ratio of the sum of within-cluster scatter to between-cluster separation. The measure is defined as:

$$DB = \frac{1}{n} \sum_{i=1, i \neq j}^n \max \left(\frac{\sigma_i + \sigma_j}{d(c_i, c_j)} \right) \quad (6)$$

where n is the number of clusters, σ_i is the average distance of all data instances in cluster i to their cluster center c_i , σ_j is the average distance of all data instances in cluster j to their cluster center c_j , and $d(c_i, c_j)$ is the distance of

¹¹ Available at <http://nl2.ijs.si/hCorpus.html#DSI>

¹² Note, however, that more elaborate clustering techniques were not used as our goal was not to obtain the best possible clustering of data but to demonstrate and assess the semi-supervised clustering methodology and to evaluate the modified constrained k-means clustering algorithm.

cluster centers c_i and c_j . Small values of the DB measure correspond to clusters that are compact and whose centers are far away from each other.

Results in Table 1 clearly show that the proposed methodology is effective. Our semi-supervised k-means algorithm outperforms other two variants of the k-means algorithm both in terms of quality of obtained clustering and the convergence rate. Moreover, our methodology also provides control of the clustering process by incorporating knowledge about the input data space at no additional cost. With the exception of the degenerated¹³ ILP-titles dataset where it needed more steps to converge (but gave better clustering), our variant of semi-supervised constrained clustering achieved the best scores. Table 1 also demonstrates the effectiveness of the visualization method used as the k-means algorithm (with the same initial centroids used in semi-supervised k-means) outperformed randomized k-means because of the visualization which enabled us to select good initial centroids - better than those selected at random in a set of 10 trials.

Table 1. Empirical evaluation of the proposed methodology and comparison of algorithms on four document corpora. Small values of the DB measure indicate good clustering. Note that the values for k-means with random initial centroids are averaged over 10 repetitions.

	k-means		rand. k-means		semi-sup. k-means	
	DB measure	#iters	DB measure	#iters	DB measure	#iters
Yahoo finance	10.2	21	10.11	23.3	9.03	9
DSI	8.33	7	8.71	9.6	7.84	7
ILP	7.15	16	7.5	10.3	6.72	10
ILP-titles	8.12	6	8.47	7.6	7.57	14

While the evaluation was carried out on only four document corpora, some general conclusions can be drawn. Firstly, least-squares meshes visualization method is able to provide enough background knowledge about the input corpora which can be used to supervise the clustering process. However, as the proposed approach is not limited to textual data, other techniques for dimensionality reduction need to be employed for other types of data. To this end, our implementation contains multi-dimensional scaling (MDS) and its faster simplification, FastMap [7]. Secondly, using our modification of the constrained k-means algorithm, it is possible to pose specific constraints on specific clusters and the results show that, backed up by the visualization, such a setup provides powerful and efficient means to semi-supervised analysis of data. Finally, the proposed solution can be used on any kind of data by potentially modifying the similarity measure¹⁴ in the visualization algorithm.

¹³ The average number of features of vectors in this dataset is only 14 which is extremely low for a textual dataset.

¹⁴ The cosine similarity measure was used to compute similarities between TF-IDF feature vectors obtained from documents.

5 Conclusions

In this paper, we presented a novel methodology for expert-guided semi-supervised data analysis which counters the identified problems. First, it allows the user to interrelate the user's knowledge of the data with the specified size constraints. This is achieved by anchoring the constraints to the specified reference points (which remain fixed in the data space) rather than to the centroids (which move). Second, for the user to be able to specify the anchor points, the data is visualized by projecting the high dimensional feature vectors onto a planar canvas. By inspecting the visualization, the user is able to identify condensed groups of data instances and place reference points into centers of such groups of instances.

The advantage of the proposed approach is that the clusters, resulting from such semi-supervised clustering process, tend to be of higher quality than clusters obtained by using the ordinary (constrained) k-means algorithm. Furthermore, the clustering optimization process tends to converge faster. Both these effects result from the fact that the user-defined reference points are a much better set of initial centroids than the randomly selected ones. As illustrated on the Yahoo Finance dataset, the least-squares meshes visualization method provides enough background knowledge about the input data for the user to effectively supervise the clustering process. Next, using our modification of the constrained k-means algorithm, it is possible to pose specific constraints to data with certain specifics (e.g. documents talking about the same/similar topic).

In conclusion, the proposed methodology presents a powerful and effective way to supervise the analysis of data. In further work, we plan to conduct experiments also on non-textual data using appropriate visualization techniques such as PCA, MDS and SOM. We expect that the proposed methodology will demonstrate its abilities even more clearly as the number of dimensions in non-textual data is typically magnitudes lower and good low-dimensional projections are much easier to obtain.

We will also improve the user interface by integrating a keyword extractor which will provide summarized information about data instances. This will greatly improve the understanding of the observed corpora and help to identify and inspect potential groups (clusters, topics) and their properties. We also plan to release the software under an open source license. Moreover, we will offer the individual components of the presented methodology as a set of services, freely available on the Web, and ready to be used in any service-oriented data mining environment.

References

1. Berkhin, P.: Survey of Clustering Data Mining Techniques. Research Paper. Accrue Software Inc. (2002)
2. Bertsekas, D.P.: Linear Network Optimization. MIT Press, Cambridge (1991)
3. Bradley, P.S., Bennett, K.P., Demiriz, A.: Constrained K-Means Clustering. Microsoft Research publication, MSR-TR-2000-65 (May 2000)

4. Bradley, P.S., Mangasarian, O.L., Street, W.N.: Clustering via concave minimization. In: *Advances in Neural Information Processing Systems*, vol. 9, pp. 368–374. MIT Press, Cambridge (1997)
5. Davies, D.L., Bouldin, D.W.: A cluster separation measure. *IEEE Trans. Pattern Anal. Machine Intelligence* 1(4), 224–227 (1979)
6. Dhillon, I., Guan, Y., Kogan, J.: Refining clusters in high dimensional data. In: *Second SIAM ICDM Workshop on Clustering High Dimensional Data* (2002)
7. Faloutsos, C., Lin, K.: FastMap: a fast algorithm for indexing, data-mining and visualization of traditional and multimedia datasets. In: *Proc. of the ACM SIGMOD Intl. Conf. on Management of Data* (1995)
8. Forgy, E.: Cluster analysis of multivariate data: Efficiency versus interpretability of classification. *Biometrics* 21, 768–780 (1965)
9. Fortuna, B., Grobelnik, M., Mladenić, D.: Semi-automatic Data-driven Ontology Construction System. In: *Proc. of the 9th Intl. Multiconf. Information Society IS 2006*, Ljubljana, Slovenia (2006)
10. Gansner, E.R., Koren, Y., North, S.: Graph Drawing by Stress Majorization. In: Pacl, J. (ed.) *GD 2004*. LNCS, vol. 3383, pp. 239–250. Springer, Heidelberg (2005)
11. Karp, R.M.: Reducibility Among Combinatorial Problems. In: Miller, R.E., Thatcher, J.W. (eds.) *Complexity of Computer Computations*, pp. 85–103. Plenum, New York (1972)
12. Kohonen, T.: *Self-Organizing Maps*. Springer, Heidelberg (1997)
13. Paige, C.C., Saunders, M.A.: Algorithm 583; LSQR: Sparse Linear Equations and Least-squares Problems. *ACM Trans. on Mathematical Software (TOMS)* 8(2), 195–209 (1982)
14. Panlovich, F.V., Nonato, L.G., Minghim, R.: Visual Mapping of Text Collections through a Fast High Precision Projection Technique. In: *Proc. of the 10th Conf. on Information Visualization*, pp. 282–290 (2006)
15. Podpečan, V., Juršič, M., Žakova, M., Lavrač, N.: Towards a Service-Oriented Knowledge Discovery Platform. In: *SoKD: ECML/PKDD 2009 workshop on Third Generation Data Mining* (2009)
16. Sorkine, O., Cohen-Or, D.: Least-squares Meshes. In: *Proc. of the Intl. Conference on Shape Modeling*, pp. 191–199 (2004)
17. Tan, P., Steinbach, M., Kumar, V.: *Introduction to Data mining*. Addison Wesley, Reading (2006)
18. Tung, A.K.H., Ng, R.T., Lakshmanan, L.V.S., Han, J.: Constraint-based clustering in large databases. In: *Proc. of the 8th Intl. Conf. on Database Theory*, pp. 405–419 (2001)
19. Wagstaff, K., Cardie, C.: Clustering with Instance-level Constraints. In: *Proc. of the 17th Intl. Conf. on Machine Learning*, pp. 1103–1110 (2000)

Partial Weighted MaxSAT for Optimal Planning

Nathan Robinson¹, Charles Gretton², Duc Nghia Pham¹, and Abdul Sattar¹

¹ ATOMIC Project, Queensland Research Lab, NICTA and
Institute for Integrated and Intelligent Systems, Griffith University, QLD, Australia

² School of Computer Science, University of Birmingham

Abstract. We consider the problem of computing optimal plans for propositional planning problems with action costs. In the spirit of leveraging advances in general-purpose automated reasoning for that setting, we develop an approach that operates by solving a sequence of *partial weighted MaxSAT* problems, each of which corresponds to a step-bounded variant of the problem at hand. Our approach is the first SAT-based system in which a proof of cost-optimality is obtained using a MaxSAT procedure. It is also the first system of this kind to incorporate an admissible planning heuristic. We perform a detailed empirical evaluation of our work using benchmarks from a number of International Planning Competitions.

1 Introduction

Recently there have been significant advances in the direction of optimal planning procedures that operate by making multiple queries to a decision procedure, usually a Boolean SAT procedure. For example, the work of Hoffman *et al.* [1] answers a key challenge from Kautz [2] by demonstrating how existing SAT-based planning techniques can be made effective solution procedures for fixed-horizon planning with metric resource constraints. In the same vein, Russell & Holden [3] and Giunchiglia & Maratea [4] develop optimal SAT-based procedures for *net-benefit* planning in fixed-horizon problems. In that case actions can have costs and goal utilities can be interdependent. Moreover, in the direction of improving the scalability and efficiency of SAT-based approaches in step-optimal (and indeed fixed-horizon) planning, Robinson *et al.* [5] presents an encoding of step-bounded planning problems that shows significant performance gains over previous results. Large performance gains have also been demonstrated where efficient and sophisticated query strategies are employed [6,7]. Summarising, in the settings of step-optimal and fixed-horizon planning, recent works have demonstrated that SAT-based techniques inspired by systems like BLACKBOX [8] continue to dominate other approaches.

Considering the planning literature more generally, numerous distinct criteria for plan optimality have been proposed. These include: (1) Minimise *makespan* (a.k.a. *step-optimality*); The objective is to find a plan of minimal length. (2) Minimise *plan cost*; Each action has a numeric cost, a plan's cost is the sum of the costs of its constituent actions, and an optimal plan has minimal cost. (3) Maximise *net-benefit*; States (resp. actions) have rewards (resp. costs), and an optimal plan is a sequence of actions executable from the starting state that induces a behaviour of maximal *utility* – These

problems are sometimes called *oversubscribed*, and were recently shown to be equivalent (using a compilation) to the cost-optimising setting [9]. One key observation to be made is that the above optimality criteria are often conflicting. For example, a plan with minimal *makespan* is not guaranteed to be *cost-* or *utility-*optimal. Indeed, in the general case there is no link between the number of plan steps (planning horizon) and plan quality.

Existing SAT-based planning procedures are limited to *makespan*-optimal and *fixed-horizon* settings – i.e., either the objective is to minimise the number of plan steps, or valid optimal solutions are constrained to be of, or less than, a fixed length. Thus, the use of SAT-based techniques is limited in practice. For example, optimal SAT-based planning procedures were unable to participate effectively at the International Planning Competition (IPC) in 2008 due to the adoption of a single optimisation criteria (cost-optimality). This paper overcomes that restriction, developing COS-P, the first sound and complete cost-optimal planning procedure based solely on a Boolean SAT(isifiability) procedure. Thus, we open the door to leveraging SAT technology in planning settings with arbitrary optimisation criteria.

The remainder of this paper is organised as follows. We first give an overview of optimal propositional planning with action costs, delete relaxations of that problem, and the partial weighted MaxSAT optimisation problem. We then describe our approach in detail, developing compilations to partial weighted MaxSAT of the fixed-horizon planning problem, and of the fixed horizon problem with a relaxed suffix. Following this we develop our novel MaxSAT solution procedure PWM-RSAT. We then empirically evaluate our approach on planning benchmarks from a number of IPCs. Finally we discuss some related work and propose some interesting directions for future research.

2 Background and Notations

2.1 Propositional Planning with Action Costs

A propositional planning problem with costs is a 5-tuple $\Pi = \langle P, \mathcal{A}, s_0, \mathcal{G}, \mathcal{C} \rangle$. Here, P is a set of propositions that characterise problem states; \mathcal{A} is the set of actions that can induce state transitions; $s_0 \subseteq P$ is the starting state; And $\mathcal{G} \subseteq P$ is the set of propositions that characterise the goal. The function $\mathcal{C} : \mathcal{A} \rightarrow \mathbb{R}_0^+$ is a bounded cost function that assigns a non-negative cost-value to each action. This value corresponds to the cost of executing the action.

Each action $a \in \mathcal{A}$ is described in terms of its preconditions $pre(a) \subseteq P$, positive effects $eff_{\bullet}(a) \subseteq P$, and negative effects $eff_{\circ}(a) \subseteq P$. An action a can be executed at a state $s \subseteq P$ when $pre(a) \subseteq s$. We write $\mathcal{A}(s)$ for the set of actions that can be executed at state s – Formally, $\mathcal{A}(s) \equiv \{a \mid a \in \mathcal{A}, pre(a) \subseteq s\}$. When $a \in \mathcal{A}(s)$ is executed at s the successive state is $(s \cup eff_{\bullet}(a)) \setminus eff_{\circ}(a)$. Actions cannot both add and delete the same proposition – i.e., $eff_{\bullet}(a) \cap eff_{\circ}(a) \equiv \emptyset$.¹ A state s is a *goal state* iff $\mathcal{G} \subseteq s$.

Usually any two actions $a_1, a_2 \in \mathcal{A}$ are permitted to be executed instantaneously in parallel at a state provided any serial execution of the actions is valid and achieves

¹ In practice this case is given a special semantics, the details of which shall not be considered further here.

an identical outcome. When two actions cannot be executed in parallel we say they *conflict*. Supposing non-conflicting actions can be executed instantaneously in parallel, a *plan* π is a discrete sequence of time-indexed sets of non-conflicting actions which, when applied to the start state, lead to a goal state. We say a plan is *serial* (a.k.a. *linear plan*), denoted π , if each time-indexed set contains one action. Finally, where A^i is the set of actions at step i of $\pi = [A^1, A^2, \dots, A^h]$, the cost of π , written $\mathcal{C}(\pi)$, is:

$$\mathcal{C}(\pi) = \sum_{i=1}^h \sum_{a \in A^i} \mathcal{C}(a)$$

A number of different conditions for plan optimality can be defined. In particular, a plan is *parallel step-optimal* if no shorter plan of the same parallel format exists. The definition for *serial step-optimality* is identical, but also respects the condition that a valid plan has only one action executed at each step. A plan π^* is *cost-optimal* if there is no plan π s.t. $\mathcal{C}(\pi) < \mathcal{C}(\pi^*)$. Finally, we draw the reader's attention to the fact that the definition of cost-optimality is not dependent on the plan format.

2.2 The Relaxed Planning Problem

A *delete relaxation* Π^+ of a planning problem Π is an equivalent problem in all respects except the definition of actions. In particular, the set of actions \mathcal{A}^+ in Π^+ comprises the elements $a \in \mathcal{A}$ from Π altered so that $\text{eff}_o(a) \equiv \emptyset$. The relaxed problem has two key properties of interest here. First, the cost of an optimal plan from any reachable state in Π is greater than or equal to the cost of the optimal plan from that state in Π^+ . Consequently relaxed planning can yield a useful admissible heuristic in search. For example, a best-first search such as A^* can be heuristically directed towards an optimal solution by using the costs of relaxed plans to arrange the priority queue. Second, although NP-hard to solve optimally in general [10], in practice optimal solutions to the relaxed problem Π^+ are more easily computed than for Π .

2.3 Partial Weighted MaxSAT

A Boolean SAT problem is a decision problem, instances of which are typically expressed as a CNF propositional formula. A CNF corresponds to a conjunction over clauses, each of which corresponds to a disjunction over literals. A literal is either a proposition (i.e., Boolean variable symbol) or its negation. Where \models denotes semantic entailment for propositional logic, a solution associated with a formula ϕ is an assignment (a.k.a. valuation) \mathcal{V} of truth values to propositions with the property $\mathcal{V} \models \phi$.

A Boolean MaxSAT problem is an optimisation problem related to SAT. In practice a problem instance is again typically expressed as a CNF, however the objective now is to compute a valuation that maximises the number of satisfied clauses. In detail, writing $\kappa \in \phi$ if κ is a clause in formula ϕ , and taking $\mathcal{V} \models \kappa$ to have numeric value 1 when valid, and 0 otherwise, a solution \mathcal{V}^* to a MaxSAT problem has the property:

$$\mathcal{V}^* = \arg \max_{\mathcal{V}} \sum_{\kappa \in \phi} (\mathcal{V} \models \kappa)$$

A *weighted* MaxSAT problem [11], denoted ψ , is a MaxSAT problem where each clause $\kappa \in \psi$ has a bounded positive numerical weight $\omega(\kappa)$. The optimal solution \mathcal{V}^* to some ψ satisfies the following equation:

$$\mathcal{V}^* = \arg \max_{\mathcal{V}} \sum_{\kappa \in \psi} \omega(\kappa) (\mathcal{V} \models \kappa)$$

Finally, the *partial* weighted MaxSAT problem [12] is a variant of weighted MaxSAT that distinguishes between *hard* and *soft* clauses. Only soft clauses are given a weight. In these problems a solution is valid iff it satisfies all hard clauses. Therefore we have a notion of satisfiability. In particular, if the *hard* problem fragment of a partial weighted MaxSAT formula is unsatisfiable, then we say the formula is unsatisfiable. The definition of satisfiable follows naturally. An optimal solution to a partial weighted MaxSAT problem is an assignment \mathcal{V}^* that is both valid and satisfies the above equation.

3 COS-P

We now describe COS-P, our planner that operates by iteratively solving variants of n -step-bounded instances of the problem at hand for successively larger n . Solutions to the intermediate step-bounded instances are obtained by compiling them into equivalent partial weighted MaxSAT problems, and then using our own MaxSAT procedure PWM-RSAT to compute their optimal solutions.

COS-P compiles and solves two variants, VARIANT-I and VARIANT-II, of the intermediate instances. Those are characterised in terms of their optimal solutions. Adopting the notation Π_n for the n -step-bounded variant of Π , VARIANT-I admits optimal solutions that correspond to minimal cost plans in the parallel format for Π_n . VARIANT-II admits optimal plans with the following structure. Each has a prefix which corresponds to n sets of actions from Π_n .² Plans can have an arbitrary length suffix (including length 0) comprised of actions from the delete relaxation Π^+ .

Both variants can be categorised as *direct*, *constructive*, and *tightly sound*. They are *direct* because we have a Boolean variable in the MaxSAT problem for every action and state proposition at each plan step. They are *constructive* because any satisfying model and its cost in the MaxSAT instances corresponds to a plan and its cost in the source problem. Critically, our compilations are *tightly sound*, in the sense that every plan with cost c in the source planning problem has a corresponding satisfying model of cost c in the MaxSAT encoding and *vice versa*. This permits two key observations about VARIANT-I and VARIANT-II. First, when both variants yield an optimal solution, and both those solutions have identical cost, then the solution to VARIANT-I is a cost-optimal plan for Π . Second, if Π is soluble, then there exists some n for which the observation of global optimality shall be made by COS-P. Finally, we have that COS-P is a sound and complete optimal planning procedure for propositional problems with action costs.

For the remainder of this section we present the compilation for VARIANT-I and VARIANT-II. In the following section we describe the MaxSAT procedure PWM-RSAT that we developed for use by COS-P.

² i.e., an n -step plan prefix in the parallel format.

3.1 VARIANT-I: Bounded Cost-Optimal Planning

We now describe a direct compilation of the bounded propositional planning problem with action costs to a partial weighted MaxSAT formula ψ . The source of our compilation is the plangraph. This is an obvious choice because *reachability* and *neededness* analysis performed during construction of the plangraph yields important mutex constraints between action and propositional variables [13]. Such constraints are not deduced independently by modern SAT procedures such as RSAT2.02 [14].

Below, we develop our compilation in terms of a list of 6 Schemata. The first 5 schemata capture the *hard* logical planning constraints, and Schema 6 reflects the action costs. Overall, the schemata we develop below make use of the following propositional variables. For each action occurring at a step $t = 0, \dots, n-1$ (excluding *noop* actions), we have a variable a^t . We define a fluent to be a state proposition whose truth value can be modified by action executions. For each fluent occurring at step $t = 0, \dots, n$ we have a variable p^t . Also, we have $make(p) \equiv \{a | a \in \mathcal{A}, p \in eff_{\bullet}(a)\}$, and $break(p) \equiv \{a | a \in \mathcal{A}, p \in eff_{\circ}(a)\}$. Below we avoid annotating variables with their time index if it is clear from the context. Lastly, all constraints are hard unless stated otherwise.

1. *Goal and start state axioms*: We have a unit clause containing p^0 for every $p \in s_0$ and $\neg p^n$ for every $p \in \mathcal{G}$.

2. *Precondition and effect axioms*: For every action a at each plan step t , we have clauses that require: (i) the action implies its precondition, (ii) the action implies its positive effects, and (iii) the action implies its negative effects:

$$[a^t \rightarrow \bigwedge_{p \in pre(a)} p^t] \wedge [a^t \rightarrow \bigwedge_{p \in eff_{\bullet}(a)} p^{t+1}] \wedge [a^t \rightarrow \bigwedge_{p \in eff_{\circ}(a)} \neg p^{t+1}]$$

3. *Mutex axioms*: For every pair of mutex symbols (actions or fluents) p_1 and p_2 at step t , we have a clause: $\neg p_1^t \wedge \neg p_2^t$

4. *At least one action axioms*: Where \mathcal{A}^t is the set of actions at step t , we have a clause that requires at least one action be executed at step t : $\bigvee_{a^t \in \mathcal{A}^t} a^t$

5. *Frame axioms*: These constrain how the truth values of fluents change over successive plan steps. For each proposition p^t , $t > 0$ we include the following clauses:

$$[p^t \rightarrow (p^{t-1} \vee \bigvee_{a \in make(p)} a^{t-1})] \wedge [\neg p^t \rightarrow (\neg p^{t-1} \vee \bigvee_{a \in break(p)} a^{t-1})]$$

6. *Action cost axioms (soft)*: Finally, we have a set of soft constraints for actions. In particular, for each action variable a^t such that $\mathcal{C}(a) > 0$, we have a unit clause $\kappa_i := \{\neg a^t\}$ with weight $\omega(\kappa_i) = \mathcal{C}(a)$.

3.2 VARIANT-II: n -Step with a Relaxed Suffix

We now describe a direct compilation of the problem Π_n from the previous section, along with the addition of a causal encoding of the delete relaxation, that we make available from step n .³ From hereon we refer to the latter as the relaxed suffix.

Our encoding of the relaxed suffix is *causal* in the sense developed in [15] for their ground parallel *causal* encoding of propositional planning in SAT. This requires additional variables to those developed for VARIANT-I. In particular, for each fluent p and relaxed action $a \in \mathcal{A}^+$ we have corresponding variables p^+ and a^+ . That p_i^+ is true intuitively means: (1) That p_i^t was false (see VARIANT-I), and (2) That $p_i \in \mathcal{G}$, or p_i^+ is

³ In VARIANT-II goal constraints from Schema 1 are omitted from Π_n .

the cause of another fluent p_j^+ in a relaxed suffix to the goal. That a^+ is true means that a is executed in the relaxed suffix. We also require a set of causal link variables. These are best introduced in terms of a recursively defined set S^∞ as follows.

$$\begin{aligned} S^0 &\equiv \{\mathcal{K}(p_i, p_j) \mid a \in \mathcal{A}^+, p_i \in \text{pre}(a), p_j \in \text{eff}_\bullet(a_i)\} \\ S^{i+1} &\equiv S^i \cup \{\mathcal{K}(p_j, p_l) \mid \mathcal{K}(p_j, p_k), \mathcal{K}(p_k, p_l) \in S^i\} \end{aligned}$$

For each $\mathcal{K}(p_i, p_j) \in S^\infty$ we have a corresponding variable. Intuitively, if proposition $\mathcal{K}(p_i, p_j)$ is true then p_i is the cause of p_j in the plan suffix.

VARIANT-II includes all schemata from VARIANT-I except the *goal axioms* of Schema 1. We also suppose Schema 6 is now inclusive of a^+ symbols. Additionally we have the following Schemata.

7. *Relaxed goal axioms*: For each fluent $p \in \mathcal{G}$ we assert that it is either achieved at the planning horizon n , or using a relaxed action in \mathcal{A}^+ . This is expressed with a clause:

$$p^n \vee p^+$$

8. *Relaxed fluent support axioms*: For each fluent p we have a clause:

$$p^+ \rightarrow (\bigvee_{a \in \text{make}(p)} a^+)$$

9. *Causal link axioms*: For all fluents p_i , taking all $a \in \text{make}(p_i)$ and $p_j \in \text{PRE}(a)$, we have the following clause:

$$(p_i^+ \wedge a^+) \rightarrow (p_j^n \vee \mathcal{K}(p_j^+, p_i^+))$$

This constraint asserts that if action a_1^+ is executed, then its preconditions must be true at horizon n , or be supported by some other action a_2^+ with $p_2 \in \text{eff}_\bullet(a_2)$.

10. *Causality implies cause and effect axiom*: For each causal link variable $\mathcal{K}(p_1^+, p_2^+)$ we have a clause:

$$\mathcal{K}(p_1^+, p_2^+) \rightarrow (p_1^+ \wedge p_2^+)$$

11. *Transitive closure and anti-reflexivity axioms*: For causal link variable $\mathcal{K}(p^+, p^+)$ we have a unit clause containing that variable negated. For pairs of causal link variables $(\mathcal{K}(p_1^+, p_2^+), \mathcal{K}(p_2^+, p_3^+))$:

$$(\mathcal{K}(p_1^+, p_2^+) \wedge \mathcal{K}(p_2^+, p_3^+)) \rightarrow \mathcal{K}(p_1^+, p_3^+)$$

12. *Only necessary relaxed fluent axioms*: For each fluent p we have a constraint:

$$\neg p^+ \vee \neg p^n$$

13. *Relaxed action cost dominance axioms*: Let \vec{P} be a set of non-mutex fluents at horizon n . Relaxed action a_1^+ is *redundant* in an optimal solution to a VARIANT-II instance, if the fluents in \vec{P} are true at horizon n and there exists a relaxed action a_2^+ such that: (1) $\text{cost}(a_2) \leq \text{cost}(a_1)$, (2) $\text{pre}(a_2) \setminus \vec{P} \subseteq \text{pre}(a_1) \setminus \vec{P}$, and (3) $\text{eff}_\bullet(a_1) \setminus \vec{P} \subseteq \text{eff}_\bullet(a_2) \setminus \vec{P}$. For relaxed action a^+ that is *redundant* for \vec{P}_1 and not redundant for any \vec{P}_2 , if $|\vec{P}_2| < |\vec{P}_1|$ we have a clause:⁴

$$(\bigwedge_{p \in \vec{P}_1} p^n) \rightarrow \neg a^+$$

⁴ In practise we limit $|\vec{P}_1|$ to 2.

The schemata we have given thus far are theoretically sufficient for our purpose. However, in a relaxed suffix most causal links are not relevant to the relaxed cost of reaching the goal from a particular state at horizon n . For example, in a logistics problem, if a truck t at location l_1 needs to be moved directly to location l_2 , then the fact that the truck is at any other location should not support it being at l_2 – i.e. $\neg \mathcal{K}(\text{at}(t, l_3), \text{at}(t, l_2)), l_3 \neq l_1$.

The following schemata provide a number of *layers* that actions and fluents in the relaxed suffix can be assigned to. Fluents and actions are forced to occur as early in the set of layers as possible and are only assigned to a layer if all supporting actions and fluents occur at earlier layers. The orderings of fluents in the relaxed layers is used to restrict the truth values of the causal link variables. The admissibility of the heuristic estimate of the relaxed suffix is independent of the number of relaxed layers.

We pick an horizon $k > n$ and generate a copy a^{+l} of each relaxed action a^+ at each layer $l \in \{n, \dots, k-1\}$ and a copy p^{+l} of each fluent p^+ at each layer $l \in \{n+1, \dots, k\}$. We also have an auxiliary variable $aux(p^{+l})$ for each fluent p^{+l} at each suffix layer $n+1, \dots, k$. Intuitively, proposition $aux(p^{+l})$ says that p is false at every layer in the relaxed suffix from n to l .⁵

14. Layered relaxed action axioms: For each layered relaxed action a^{+l} we have a clause:

$$a^{+l} \rightarrow a^+$$

15. Layered relaxed actions only once axioms: For each relaxed action a^+ and pair of layers $l_1, l_2 \in \{n, \dots, k-1\}$, where $l_1 \neq l_2$, we have:

$$\neg a^{+l_1} \vee \neg a^{+l_2}$$

16. Layered relaxed action precondition axioms: For each layered relaxed action a^{+l_1} we have a set of clauses:

$$a^{+l_1} \rightarrow \bigwedge_{p \in \text{PRE}(a)} \bigvee_{l_2 \in \{n, \dots, l_1\}} p^{+l_2}$$

17. Layered relaxed action effect axioms: For each layered relaxed action a^{+l_1} and $p \in \text{ADD}(a)$ there is a clause:

$$(a^{+l_1} \wedge p^+) \rightarrow \bigvee_{l_2 \in n+1, \dots, l_1+1} p^{+l_2}$$

18. Layered relaxed action as early as possible axioms: For each layered relaxed action a^{+l_1} , if $l_1 = n$, we have a clause:

$$a^+ \rightarrow \bigvee_{p \in \text{PRE}(a)} \neg p^n \vee a^{+n}$$

if $l_1 > n$, we add:

$$a^+ \rightarrow \bigvee_{l_2 \in n, \dots, l_1-1} a^{+l_2} \vee \bigvee_{p \in \text{PRE}(a)} aux(p^{+l_1}) \vee a^{+l_1}$$

19. Auxiliary variable axioms: For each auxiliary variable $aux(p^{+l_1})$ there is a set of clauses:

$$aux(p^{+l_1}) \longleftrightarrow (p^n \wedge \bigwedge_{l_2 \in \{n+1, \dots, l_1\}} \neg p^{+l_2})$$

⁵ There are no cost constraints associated with the layered copies of relaxed action variables.

20. *Layered fluent axioms*: For each layered fluent p^{+l} we add:

$$p^{+l} \rightarrow p^+$$

21. *Layered fluent frame axioms*: For each layered fluent p^{+l} there is a clause:

$$p^{+l} \rightarrow \bigvee_{a \in \text{make}(p)} a^{+l-1}$$

22. *Layered fluent as early as possible axioms*: For each layered fluent p^{+l_1} there is a set of clauses:

$$p^{+l_1} \rightarrow \bigwedge_{a \in \text{make}(p)} \bigwedge_{l_2 \in n, \dots, l_1-2} \neg a^{+l_2}$$

23. *Layered fluent only once axioms*: For each fluent p and pair of layers $l_1, l_2 \in \{n+1, \dots, k\}$, where $l_1 \neq l_2$, there is a clause:

$$\neg p^{+l_1} \vee \neg p^{+l_2}$$

24. *Layered fluents prohibit causal links axioms*: For each layered fluent $p_1^{+l_1}$ and fluent p_2 such that $p_1 \neq p_2$ and $\exists \mathcal{K}(p_2^+, p_1^+)$ there is a clause:

$$p_1^{+l_1} \rightarrow (\bigvee_{l_2 \in \{n+1, \dots, l_1-1\}} p_2^{+l_2} \vee \neg \mathcal{K}(p_2^+, p_1^+))$$

4 PWM-RSAT

We find that branch-and-bound procedures for *partial weighted MaxSAT* [11,12] are ineffective at solving our direct encodings of bounded planning problems. Thus, taking the RSAT2.02 codebase as a starting point, we developed PWM-RSAT, a more efficient optimisation procedure for this setting. An outline of the algorithm is given in Algorithm 1. Based on RSAT [16], PWM-RSAT can broadly be described as a backtracking search with Boolean unit propagation. It features common enhancements from state-of-the-art SAT solvers, including conflict driven *clause learning* with *non-chronological* backtracking [17,18], and *restarts* [19].

Algorithm 1 outlines two variants of PWM-RSAT for solving VARIANT-I and VARIANT-II formulas: lines 5-6 will only be invoked if the input formula is a VARIANT-II encoding. These lines prevent the solver from exploring assignments implying that the same state occurs at more than one planning layer.

Apart from the above difference, the two variants of PWM-RSAT work as follows. At the beginning of the search, the current partial assignment \mathcal{V} of truth values to variables in ψ is set to empty and its associated cost c is set to 0. We use \hat{c} to track the best result found so far for the minimum cost of satisfying ψ^∞ given ψ^+ . \mathcal{V}^* is the total assignment associated with \hat{c} . Initially, \mathcal{V}^* is empty and \hat{c} is set to an input non-negative weight bound \hat{c}^I (if none is known then $\hat{c} = \hat{c}^I := \infty$). Note that the set of *asserting clauses* Γ is initiated to empty as no clauses have been learnt yet.

The solver then repeatedly tries to expand the partial assignment \mathcal{V} until either the optimal solution is found or ψ is proved unsatisfiable (line 4-21). At each iteration, a call to $\text{SatUP}(\mathcal{V}, \psi, \kappa)$ applies unit propagation to a unit clause $\kappa \in \psi$ and adds new variable assignments to \mathcal{V} . If κ is not a unit clause, $\text{SatUP}(\mathcal{V}, \psi, \kappa)$ returns 1 if κ is satisfied by \mathcal{V} , and 0 otherwise. The current cost c is also updated (line 7). If $c \geq \hat{c}$, then the solver will perform a backtrack-by-cost to a previous point where $c < \hat{c}$ (line 8-9).

Algorithm 1. Cost-Optimal RSat — PWM-RSAT

```

1: Input:
   - A given non-negative weight bound  $\hat{c}^I$ . If none is known:  $\hat{c}^I := \infty$ 
   - A CNF formula  $\psi$  consists of the hard clause set  $\psi^\infty$  and the soft clause set  $\psi^+$ 
2:  $c \leftarrow 0$ ;  $\hat{c} \leftarrow \hat{c}^I$ ;
3:  $\mathcal{V}, \mathcal{V}^* \leftarrow []$ ;  $\Gamma \leftarrow \emptyset$ ;
4: while true do
5:   if solving Variant-II && duplicating-layers( $\mathcal{V}$ ) then
6:     pop elements from  $\mathcal{V}$  until  $\neg$ duplicating-layers( $\mathcal{V}$ ); continue;
7:    $c \leftarrow \sum_{\kappa \in \psi^+} \omega(\kappa) \text{SatUP}(\mathcal{V}, \psi, \kappa)$ ;
8:   if  $c \geq \hat{c}$  then
9:     pop elements from  $\mathcal{V}$  until  $c < \hat{c}$ ; continue;
10:  if  $\exists \kappa \in (\psi^\infty \wedge \Gamma)$  s.t.  $\neg \text{SatUP}(\mathcal{V}, \psi^\infty \wedge \Gamma, \kappa)$  then
11:    if restart then  $\mathcal{V} \leftarrow []$ ; continue;
12:    learn clause with assertion level  $m$ ; add it to  $\Gamma$ ;
13:    pop elements from  $\mathcal{V}$  until  $|\mathcal{V}| = m$ ;
14:    if  $\mathcal{V} = []$  then
15:      if  $\mathcal{V}^* \neq []$  then return  $(\mathcal{V}^*, \hat{c})$  as the solution;
16:      else return UNSATISFIABLE;
17:  else
18:    if  $\mathcal{V}$  is total then
19:       $\mathcal{V}^* \leftarrow \mathcal{V}$ ;  $\hat{c} \leftarrow c$ ;
20:      pop elements from  $\mathcal{V}$  until  $c < \hat{c}$ ;
21:      add a new variable assignment to  $\mathcal{V}$ ;

```

During the search, if the current assignment \mathcal{V} violates any clause in $(\psi^\infty \wedge \Gamma)$, then the solver will either (i) restart if required (line 11), or (ii) try to learn the conflict (line 12) and then backtrack (line 13). If the backtracking causes all assignments in \mathcal{V} to be undone, then the solver has successfully proved that either (i) (\mathcal{V}^*, \hat{c}) is the optimal solution, or (ii) ψ is unsatisfiable if \mathcal{V}^* remains empty (line 14-16). Otherwise, if \mathcal{V} does not violate any clause in $(\psi^\infty \wedge \Gamma)$ (line 17), then the solver will heuristically add a new variable assignment to \mathcal{V} (line 21) and repeat the loop in line 4. Note that if \mathcal{V} is already complete, the better solution is stored in \mathcal{V}^* together with the new lower cost \hat{c} (line 19). The solver also performs a backtrack by cost (line 20) before trying to expand \mathcal{V} in line 21.

5 Experimental Results

We implemented both COS-P and PWM-RSAT in C++. We now discuss our experimental comparison of COS-P with IPC baseline planner BASELINE,⁶ and a version of COS-P called H-ORACLE. The latter is given (by an oracle) the shortest horizon that yields a globally optimal plan. Planning benchmarks included in our evaluation include: IPC-6: ELEVATORS, PEG SOITAIRE, and TRANSPORT; IPC-5: STORAGE, and TPP; IPC-3: DEPOTS, DRIVERLOG, FREECCELL, ROVERS, SATELLITE, and ZENOTRAVEL; and IPC-1: BLOCKS, GRIPPER, and MICONIC. We also developed our own domain, called FTB, that demonstrates the effectiveness of the factored problem representations employed by SAT-based systems such as COS-P. This domain has the following important properties: (1) it has exponentially many states in the number of problem objects, (2) if there are n objects, then the branching factor is such that a breadth-first search

⁶ The *de facto* winning entry at the last IPC.

encounters all the states at depth n , and (3) all plans have length n , and plan optimality is determined by the first and last actions (only) of the plan. This domain cripples state-based systems such as HSP, BASELINE, and GAMER, either because they are doing a non-factored forward heuristic search, or because —i.e., in the case of GAMER and BASELINE— they perform a breadth-first search. Finally, experiments were run on a cluster of AMD Opteron 252 2.6GHz processors, each with 2GB of RAM. All plans computed by COS-P, H-ORACLE, and BASELINE were verified by the Strathelyde Planning Group plan verifier VAL, and computed within a timeout of 30 minutes.

The results of our experiments are summarised in Table 1. For each domain there is one row for the hardest problem instance solved by each of the three planners. Here, we measure problem hardness as the time it takes each solver to return the optimal plan. In some domains we also include additional instances. Using the same experimental data as for Table 1, Figure 1 plots the cumulative number of instances solved over time by each planning system, supposing invocations of the systems on problem instances are made in parallel. It is important to note that the size of the CNF encodings required by COS-P (and H-ORACLE) are not prohibitively large — i.e., where the SAT-based approaches fail, this is typically because they exceed the 30 minutes timeout, and not because they exhaust system memory.

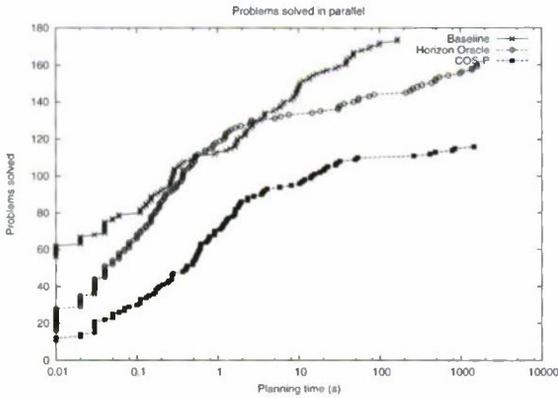


Fig. 1. The number of problems solved in parallel after a given planning time for each approach

COS-P outperforms the BASELINE in the BLOCKS and FTB domains. For example, on BLOCKS-18 BASELINE takes 39.15 seconds while COS-P takes only 3.47 seconds. In other domains BASELINE outperforms COS-P, sometimes by several orders of magnitude. For example, on problem ZENOTRAVEL-4 BASELINE takes 0.04 seconds while COS-P takes 841.2. More importantly, we discovered that it is relatively easy to find a cost-optimal solution compared to proving its optimality. For example, on MICONIC-23 COS-P took 0.53 seconds to find the optimal plan but spent 1453 seconds proving cost-optimality. More generally, this observation is indicated by the performance of H-ORACLE.

Overall, we find that clause learning procedures in PWM-RSAT cannot exploit the presence of the *very* effective delete relaxation heuristic from H^+ . Consequently, a serious bottleneck of our approach comes from the time required to solve VARIANT-II

Table 1. C^* is the optimal cost for each problem. All times are in seconds. For BASELINE t is the solution time. For H-ORACLE, n is the horizon returned by the oracle and t is the time taken to find the lowest cost plan at n . For COS-P, t_t is the total time for all SAT instances, t_π is the total time for all SAT instances where the system was searching for a plan, while t_* is the total time for all SAT instances where the system is performing optimality proofs. '-' indicates that a solver either timed out or ran out of memory.

Problem	C^*	BASELINE	H-ORACLE		COS-P			
		t	n	t	n	t_t	t_π	t_*
blocks-17	28	39.83	28	0.59	28	3.61	3.61	0
blocks-18	26	39.15	26	0.53	26	3.47	3.47	0
blocks-23	30	-	30	4.61	30	32.11	32.11	0
blocks-25	34	-	34	3.43	34	29.49	29.49	0
depots-7	21	98.08	11	64.79	-	-	-	-
driverlog-3	12	0.11	7	0.043	7	484.8	0.08	484.7
driverlog-6	11	9.25	5	0.046	-	-	-	-
driverlog-7	13	100.9	7	1.26	-	-	-	-
elevators-2	26	0.33	3	0.01	3	14	0.01	13.99
elevators-5	55	167.9	-	-	-	-	-	-
elevators-13	59	28.59	10	378.6	-	-	-	-
freecell-4	26	47.36	-	-	-	-	-	-
ftb-17	401	38.28	17	0.08	17	0.27	0.09	0.18
ftb-30	1001	-	25	0.7	25	1.95	0.7	1.24
ftb-38	601	-	33	0.48	33	1.65	0.49	1.15
ftb-39	801	-	33	0.7	33	2.35	0.67	1.69
grripper-1	11	0	7	0.02	7	15.7	0.14	15.56
grripper-3	23	0.05	15	34.23	-	-	-	-
grripper-7	47	73.95	-	-	-	-	-	-
miconic-17	13	0	11	0.07	11	785.4	0.30	785.1
miconic-23	15	0.04	10	0.12	10	1454	0.53	1453
miconic-33	22	2.19	17	2.17	-	-	-	-
miconic-36	27	9.62	22	1754	-	-	-	-
miconic-39	28	10.61	24	484.1	-	-	-	-
pegsol-7	3	0	12	0.08	12	1.63	0.23	1.41
pegsol-9	5	0.02	15	7.07	15	416.6	12.25	404.4
pegsol-13	9	0.14	21	1025	-	-	-	-
pegsol-26	9	42.44	-	-	-	-	-	-
rovers-3	11	0.02	8	0.1	8	53.21	0.08	53.13
rovers-5	22	164.1	8	69.83	-	-	-	-
satellite-1	9	0	8	0.08	8	0.92	0.1	0.82
satellite-2	13	0.01	12	0.23	-	-	-	-
satellite-4	17	6.61	-	-	-	-	-	-
storage-7	14	0	14	0.45	14	1.16	1.16	0
storage-9	11	0.2	9	643.2	-	-	-	-
storage-13	18	3.47	18	112.1	18	262.8	262.8	0
storage-14	19	60.19	-	-	-	-	-	-
TPP-5	19	0.15	7	0.01	-	-	-	-
transport-1	54	0	5	0.02	5	0.27	0.03	0.24
transport-4	318	47.47	-	-	-	-	-	-
transport-23	630	0.92	9	1.28	-	-	-	-
zenotravel-4	8	0.04	7	1.07	7	843.7	2.47	841.2
zenotravel-6	11	8.77	7	54.35	-	-	-	-
zenotravel-7	15	5.21	8	1600	-	-	-	-

instances. On a positive note, those proofs are possible, and in domains such as BLOCKS and FTB, where the branching factor is high and useful plans long, the factored problem representations and corresponding solution procedures in the SAT-based setting payoff. Moreover, in fixed-horizon cost-optimal planning, the SAT approach continues to show good performance characteristics in many domains.

6 Concluding Remarks

In this paper we demonstrate that a general theorem-proving technique, particularly a DPLL procedure for Boolean SAT, can be modified to find cost-optimal solutions to propositional planning problems encoded as SAT.⁷ In particular, we modified SAT solver RSAT2.02 to create PWM-RSAT, an effective partial weighted MaxSAT procedure for problems where all *soft* constraints are unit clauses. This forms the underlying optimisation procedure in COS-P, our cost-optimal planning system that, for successive horizon lengths, uses PWM-RSAT to establish a candidate solution at that horizon, and then to determine if that candidate is globally optimal. Each candidate is a minimal cost step-bounded plan for the problem at hand. That a candidate is globally optimal is known if no step-bounded plan with a relaxed suffix has lower cost. To achieve that, we developed a MaxSAT encoding of bounded planning problems with a relaxed suffix. This constitutes the first application of causal representations of planning in propositional logic [15].

Existing work directly related to COS-P includes the hybrid solver CO-PLAN [20] and the fixed-horizon optimal system PLAN-A. Those systems placed 4th and last respectively out of 10 systems at IPC-6. CO-PLAN is hybrid in the sense that it proceeds in two phases, each of which applies a different search technique. The first phase is SAT-based, and identifies the least costly step-optimal plan. PLAN-A also performs that computation, however assumes that a least cost step-optimal plan is globally optimal – Therefore PLAN-A was not competitive because it could not find globally optimal solutions, and thus forfeited in many domains. The first phase of CO-PLAN and the PLAN-A system can be seen as more general and efficient versions of the system described in [21]. The second phase of CO-PLAN breaks from the planning-as-SAT paradigm. It corresponds to a cost-bounded anytime best-first search. The cost bound for the second phase is provided by the first phase. Although competitive with a number of other competition entries, CO-PLAN is not competitive in IPC-6 competition benchmarks with the BASELINE – The *de facto* winning entry, a brute-force A^* in which the distance-plus-cost computation always takes the distance to be zero.

Other work related to COS-P leverages SAT modulo theory (SMT) procedures to solve problems with metric resource constraints [22]. SMT-solvers typically interleave calls to a *simplex* algorithm with the *decision steps* of a backtracking search, such as DPLL. Solvers in this category include the systems LPSAT [22], TM-LPSAT [23], and NUMREACH/SMT [1]. SMT-based planners also operate according to the BLACK-BOX scheme, posing a series of step-bounded decision problems to an SMT solver until an optimal plan is achieved. Because they are not globally optimal, existing SMT systems are not directly comparable to COS-P.

The most pressing item for future work is a technique to exploit SMT —and/or branch-and-bound procedures from weighted MaxSAT— in proving the optimality of candidate solutions that PWM-RSAT yields in bounded instances. We should also exploit recent work in using useful admissible heuristics for state-based search when evaluating whether horizon n yields an optimal solution [24].

⁷ This was supposed to be possible, although in a very impractical sense (final remarks of [4]).

Acknowledgements. NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program. This work was also supported by EC FP7-IST grant 215181-CogX.

References

1. Hoffmann, J., Gomes, C.P., Selman, B., Kautz, H.A.: Sat encodings of state-space reachability problems in numeric domains. In: Proc. IJCAI (2007)
2. Kautz, H.A.: Deconstructing planning as satisfiability. In: Proc. AAAI (2006)
3. Russell, R., Holden, S.: Handling goal utility dependencies in a satisfiability framework. In: Proc. ICAPS (2010)
4. Giunchiglia, E., Maratea, M.: Planning as satisfiability with preferences. In: Proc. ICAPS (2007)
5. Robinson, N., Gretton, C., Pham, D.N., Sattar, A.: Sat-based parallel planning using a split representation of actions. In: Proc. ICAPS (2009)
6. Streeter, M., Smith, S.: Using decision procedures efficiently for optimization. In: Proc. ICAPS (2007)
7. Rintanen, J.: Evaluation strategies for planning as satisfiability. In: Proc. ECAI (2004)
8. Kautz, H., Selman, B.: Unifying SAT-based and graph-based planning. In: Proc. IJCAI (1999)
9. Keyder, E., Geffner, H.: Soft goals can be compiled away. *Journal of Artificial Intelligence Research* 36(1) (2009)
10. Bylander, T.: The computational complexity of propositional strips planning. *Artificial Intelligence* 69, 165–204 (1994)
11. Argelice, J., Li, C.M., Manyá, F., Planes, J.: The first and second max-sat evaluations. *Journal on Satisfiability, Boolean Modeling and Computation* 4, 251–278 (2008)
12. Fu, Z., Malik, S.: On solving the partial max-sat problem. In: Biere, A., Gomes, C.P. (eds.) *SAT 2006*. LNCS, vol. 4121, pp. 252–265. Springer, Heidelberg (2006)
13. Blum, A., Furst, M.: Fast planning through planning graph analysis. *Artificial Intelligence* (90), 281–300 (1997)
14. Rintanen, J.: Planning graphs and propositional clause learning. In: Proc. KR (2008)
15. Kautz, H., McAllester, D., Selman, B.: Encoding plans in propositional logic. In: Proc. KR (1996)
16. Pipatsrisawat, K., Darwiche, A.: Rsat 2.0: SAT solver description. Technical Report D-153, Automated Reasoning Group, Computer Science Department, UCLA (2007)
17. Moskewicz, M.W., Madigan, C.F., Zhao, Y., Zhang, L., Malik, S.: Chaff: Engineering an Efficient SAT Solver. In: Proc. DAC (2001)
18. Marques-Silva, J.P., Sakallah, K.A.: Grasp - a new search algorithm for satisfiability. In: Proc. ICCAD (1996)
19. Huang, J.: The effect of restarts on the efficiency of clause learning. In: Proc. IJCAI (2007)
20. Robinson, N., Gretton, C., Pham, D.N.: Co-plan: Combining SAT-based planning with forward-search. In: Proc. IPC-6 (2008)
21. Büttner, M., Rintanen, J.: Satisfiability planning with constraints on the number of actions. In: Proc. ICAPS (2005)
22. Wolfman, S.A., Weld, D.S.: The LPSAT engine and its application to resource planning. In: Proc. IJCAI (1999)
23. Shin, J.A., Davis, E.: Processes and continuous change in a sat-based planner. *Artif. Intell.* 166(1-2), 194–253 (2005)
24. Helmert, M., Domshlak, C.: Landmarks, critical paths and abstractions: What's the difference anyway? In: Proc. ICAPS (2009)

Efficient Estimation of Cumulative Influence for Multiple Activation Information Diffusion Model with Continuous Time Delay

Kazumi Saito¹, Masahiro Kimura², Kouzou Ohara³, and Hiroshi Motoda⁴

¹ School of Administration and Informatics, University of Shizuoka
52-1 Yada, Suruga-ku, Shizuoka 422-8526, Japan
k-saito@u-shizuoka-ken.ac.jp

² Department of Electronics and Informatics, Ryukoku University
Otsu 520-2194, Japan
kimura@rins.ryukoku.ac.jp

³ Department of Integrated Information Technology, Aoyama Gakuin University
Kanagawa 229-8558, Japan
ohara@it.aoyama.ac.jp

⁴ Institute of Scientific and Industrial Research, Osaka University
8-1 Mihogaoka, Ibaraki, Osaka 567-0047, Japan
motoda@ar.sanken.osaka-u.ac.jp

Abstract. We show that the node cumulative influence for a particular class of information diffusion model in which a node can be activated multiple times, i.e. Susceptible/Infective/ Susceptible (SIS) Model, can be very efficiently estimated in case of independent cascade (IC) framework with asynchronous time delay. The method exploits the property of continuous time delay within a stochastic framework and analytically derives the iterative formula to estimate cumulative influence without relying on awfully lengthy simulations. We show that it can accurately estimate the cumulative influence with much less computation time (about 2 to 6 orders of magnitude less) than the naive simulation using three real world social networks and thus it can be used to rank influential nodes quite effectively. Further, we show that the SIS model with a discrete time step, i.e. fixed synchronous time delay, gives adequate results only for a small time span.

1 Introduction

The proliferation of emails, blogs and social networking services (SNS) in the World Wide Web has accelerated the creation of large social networks [1,2,3,4,5]. Social networks naturally mediate the spread of various information. Innovation, topics and even malicious rumors can propagate in the form of so-called “word-of-mouth” communications. Thus, it is now understood that social networks provide rich sources of information that is useful to help understand the dynamics of our society, e.g. who are the best group of people to spread the desired information, how people respond to other people’s opinion, what kind of topics propagate faster, how the public opinions are formed, how the way the information spreads differ from community to community, etc.

Several models have been proposed that simulate information diffusion through a network. The most widely-used model is the *independent cascade (IC)*. This is a fundamental probabilistic model of information diffusion [6,7], which can be regarded as the so-called *susceptible/infective/recovered (SIR) model* for the spread of a disease [2]. This model has been used to solve such problems as the *influence maximization problem* which is to find a limited number of nodes that are influential for the spread of information [7,8] and the *influence minimization problem* which is to suppress the spread of undesirable information by blocking a limited number of links [9]. Here, it is noted that the influence of a node is defined as the expected number of nodes that it can activate due to the stochastic nature of the information diffusion. The SIR model assumes that a node, once infected, never re-infected after it has been cured (recovered). Thus, the influence is normally defined as the expected number of recovered nodes at the end of the time span in consideration. The other class of model for the spread of a disease is the so-called *susceptible/infective/susceptible (SIS) model* [2], where a node, once infected, moves to a susceptible state and can be re-activated multiple times. A similar problem can be solved for this model, too [10,11]. In these models, efficient methods of estimating the influence have been proposed based on bond percolation, strongly connected component decomposition, burnout and pruning [8,11], but no analytical solutions have been found. Thus, efficiency remains that the computation time is 2 or 3 orders of magnitude faster than naive simulation.

The IC model above, whether it is used in SIR or SIS setting, cannot handle time-delays that are asynchronous and continuous for information propagation. Time step is incremented discretely and thus the node states are updated synchronously, which can be viewed that the time delay is fixed and synchronous. We call this “fixed time delay” for short. In reality, time flows continuously and thus information, too, propagates on this continuous time axis. For any node, information must be received at any time from any other nodes and must be allowed to propagate to yet other nodes at any other time with a possible delay, both in an asynchronous way. We call this “continuous time delay” for short. For example, the following scenario in case of SIS setting explains this need. Suppose a person A posted an article to a blog and a person B read it and responded a week later. Another person C posted an article on the same topic the next day A posted and B read it and responded the same day. B was activated twice, first by C and next by A although the time A was activated is earlier than C. Thus, for a realistic behavior analysis of information diffusion, we need to adopt a model that explicitly represents continuous asynchronous time delay. The continuous time delay SIR model was discussed in the machine learning problem setting in which the objective was to learn the parameters in the diffusion model from the observed time stamped node activation sequence data [3,12]. In [12] it was shown that the parameters can be learned by maximizing the likelihood of the observed data being produced by the model. Note that there is no need to do simulation to obtain the influence degree in case of SIR setting because the final influence degree is equal to that of the model without time delay¹ since a node is not allowed to be re-activated multiple times.

In this paper, we address the problem of efficiently estimating the *cumulative influence* of a node in the network by adopting the information diffusion model that allows

¹ This is equivalent to fixed time delay in discrete time setting.

continuous time delay and multiple activation of the same node under the framework of independent cascade model, called CTSIS for short. Interestingly, although the model we considered in this paper is most complicated among the series of the models discussed above, it is possible to derive a formula analytically, under a simplified condition, that can iteratively estimate the *cumulative influence* of a node exploiting the property of continuous time delay within a stochastic framework. What makes the analysis easier is that in case of the continuous time there is only one single node that can be activated at a time, i.e, no multiple activations at different nodes at the same time, and no simultaneous activations of a node by its multiple active parents each of which has been activated at a different time in the past. Thus it does not make sense to define the node influence at a specific time and in light of SIS and continuous time delay we naturally define the influence to be an integral over a specified time span (*cumulative influence*), which is more meaningful in many practical settings.

We show that the proposed method (called *iterative method*) can accurately estimate the cumulative influence with much less computation time (about 2 to 6 orders of magnitude less) than empirical mean of the *naive simulation method* with a limited number of runs using three real world social networks with different sizes and connectivities. The method can be used to rank influential nodes quite effectively. We compare the proposed methods with two other methods, the SIS with fixed time delay and the one which is the extreme case of the propose method where the time span is set to be infinitely large (called *infinite iterative method*). We show that these are indeed less accurate and discuss under which conditions these work well, e.g. SIS with fixed time delay only works well for a small time span.

The paper is organized as follows. We revisit the information diffusion model, in particular SIS family, in section 2, and explain the proposed method of cumulative influence estimation in section 3. Then we report the experimental results in section 4, followed by discussion in section 5. We summarize our conclusion in section 6.

2 Information Diffusion Model

Let $G = (V, E)$ be a directed network, where V and $E (\subset V \times V)$ stand for the sets of all the nodes and (directed) links, respectively. For any $v \in V$, let $\Gamma(v; G)$ denote the set of the child nodes (directed neighbors) of v , that is,

$$\Gamma(v; G) = \{w \in V; (v, w) \in E\}.$$

We consider information diffusion models on G in the susceptible/infected/susceptible (SIS) framework. In this context, infected nodes mean that they have just adopted the information, and we call these infected nodes *active* nodes.

2.1 Basic SIS Model

We first define the basic SIS model for information diffusion on G . In the model, the diffusion process unfolds in discrete time-steps $t \geq 0$, and it is assumed that the state of a node is either active or inactive. For every link $(u, v) \in E$, we specify a real value $\kappa_{u,v}$ with $0 < \kappa_{u,v} < 1$ in advance. Here, $\kappa_{u,v}$ is referred to as the *diffusion parameter*

through link (u, v) . Given an initial active node v_0 and a time span T , the diffusion process proceeds in the following way. Suppose that node u becomes active at time-step t ($< T$). Then, node u attempts to activate every $v \in I(u; G)$, and succeeds with probability $\kappa_{u,v}$. If node u succeeds, then node v will become active at time-step $t + 1$. Thus, as mentioned in 1, we can view this as synchronous fixed time delay². If multiple active nodes attempt to activate node v in time-step t , then their activation attempts are sequenced in an arbitrary order. On the other hand, node u will become inactive at time-step $t + 1$ unless it is activated by an active node in time-step t . The process terminates if the current time-step reaches the final time T .

2.2 Continuous-Time SIS Model

Next, we extend the basic SIS model so as to allow continuous-time delays, and refer to the extended model as the *continuous-time SIS (CTSIS) model*³. This model can be interpreted as *susceptible/exposed/infective/susceptible (SEIS) model* in that a node does not become active (infected) instantly when activated, but wait for a while (exposed) before it gets activated (infected). Once it gets activated, it instantly turns into susceptible state. In terms of information diffusion of some topic in blog space, this activation corresponds to posting a blog article on the topic (instantaneous action).

In the CTSIS model on G , for each link $(u, v) \in E$, we specify real values $r_{u,v}$ and $\kappa_{u,v}$ with $r_{u,v} > 0$ and $0 < \kappa_{u,v} < 1$ in advance. We refer to $r_{u,v}$ and $\kappa_{u,v}$ as the *time-delay parameter* and the *diffusion parameter* through link (u, v) , respectively.

Let T be the time span. The diffusion process unfolds in continuous-time t , and proceeds from a given initial active node v_0 in the following way. Suppose that a node u becomes active at time t ($< T$). Then a delay-time δ is chosen for u 's every child node $v \in I(u; G)$ from the exponential distribution with parameter $r_{u,v}$. If $t + \delta \leq T$, v is activated by u with success probability $\kappa_{u,v}$ at $t + \delta \leq T$. Under the continuous time framework, there is no possibility that multiple parent nodes of v simultaneously activate v exactly at the same time $t + \delta$. The process terminates if the current time reaches the final time T .

2.3 Influence Function

Let T be the time span for the CTSIS model on G . We consider a time-interval $[T_0, T_1]$ with $0 \leq T_0 < T_1 \leq T$. For any node $v \in V$, let $S(v; T_0, T_1)$ denote the total number of nodes activated within time-interval $[T_0, T_1]$ for the probabilistic diffusion process from an initial active node v under the CTSIS model. Note that $S(v; T_0, T_1)$ is a random variable. Let $\sigma(v; T_0, T_1)$ denote the expected value of $S(v; T_0, T_1)$. We call $\sigma(v; T_0, T_1)$ the *cumulative influence degree* of node v within time-interval $[T_0, T_1]$. Note that σ is a function defined on V . We call the function $\sigma(\cdot; T_0, T_1) : V \rightarrow \mathbf{R}$ the *cumulative influence function* for the CTSIS model within time-interval $[T_0, T_1]$ on network G .

It is important to estimate the cumulative influence function $\sigma(\cdot; T_0, T_1)$ efficiently. In theory we can simply estimate it by simulating the CTSIS model in the following

² This may well be called as “no time delay” because time delay is not explicitly represented in the formulation.

³ Note that the information propagates at a certain time point, but its delay can be continuous.

way. First, a sufficiently large positive integer M is specified. For each $v \in V$, the diffusion process of the CTSIS model is simulated from initial active node v , and the total number of nodes activated within time-interval $[T_0, T_1]$, $S(v; T_0, T_1)$, is calculated. Then, $\sigma(v; T_0, T_1)$ is estimated as the empirical mean of $S(v; T_0, T_1)$ that are obtained from M such simulations. We refer to this estimation method as the *naive simulation method*. However, as shown in the experiments, this is extremely inefficient, and cannot be practical (out of question). In this paper, we deal with the case “ $T_0 = 0, T_1 = T$ ” for simplicity, and we denote $\sigma(v; 0, T)$ by $\sigma(v; T)$.

3 Estimation Methods

For a given directed graph $G = (V, E)$, we identify each node with a unique integer from 1 to $|V|$. Then we can define the adjacency matrix $A \in \{0, 1\}^{|V| \times |V|}$ by setting $a_{u,v} = 1$ if $(u, v) \in E$; otherwise $a_{u,v} = 0$. We also define the probability matrix $P \in [0, 1]^{|V| \times |V|}$ by replacing each element $a_{u,v}$ to the corresponding diffusion probability $\kappa_{u,v}$ if $(u, v) \in E$. Let $f_v \in \{0, 1\}^{|V|}$ be a vector whose v -th element is 1 and other elements are 0, and $\mathbf{1} \in \{1\}^{|V|}$ be a vector whose elements are all 1.

3.1 Infinite Iterative Method

We can calculate the number of nodes that are reachable with J -steps starting from a node v by $f_v^T A^J \mathbf{1}$. Thus, when considering the diffusion probabilities, we can calculate the vector of the expected number of reachable nodes starting from each node within J steps by $P\mathbf{1} + \dots + P^J \mathbf{1}$. Therefore, in case that the time-interval is $[0, \infty]$, according to the definition of the CTSIS model, we obtain the cumulative influence degree σ_∞ as follows:

$$\sigma_\infty = \sum_{J=1}^{\infty} P^J \mathbf{1}, \quad (1)$$

Note that the vector σ_∞ consists of values of the cumulative influence functions, i.e., $\sigma(\cdot; \infty)$. We refer to this estimation method as the *infinite iterative method*.

However, there exist some intrinsic limitations to the simple iterative method, i.e., we cannot specify arbitrary time-interval $[T_0, T_1]$ and diffusion probabilities for this method. As for the diffusion probabilities, when the largest eigenvalue of the probability matrix P is less than 1, we can guarantee to obtain finite value of σ_∞ . In a simple case that the diffusion parameters are uniform for any link, i.e., $\kappa_{u,v} = \kappa$ for any $(u, v) \in E$, since the probability matrix P is equivalent to κA , the diffusion parameter κ must be less than the reciprocal of the largest eigenvalue of the adjacency matrix A . Incidentally, the calculation formula for this simple case is quite similar to that of Bonacich's centrality [13] and identical to that of Katz's measure [14].

3.2 Proposed Method

We want to estimate the cumulative influence degree within time-interval $[T_0, T_1]$ for arbitrary diffusion probabilities. To this end, we introduce the probability $R(J; T_0, T_1)$

that diffusion takes J -steps within this time-interval according to the CTSIS model. Here, in order to simplify our derivation, we focus on the simplest case that the time-delay parameters are uniform for any link, i.e., $r_{u,v} = r$ for any $(u, v) \in E$, although our approach can be naturally extended to more complex settings. In a special case where $T_0 = 0$ and $T_1 = T$, we denote this probability by $R(J; T)$. Here we note that $R(J; T_0, T_1) = R(J; T_1) - R(J; T_0)$. Thus we focus on calculation of $R(J; T)$.

Let δ_j be a random variable of a time-delay for the j -th step ($1 \leq j \leq J$). In order to meet the condition that the diffusion takes J -steps within time-interval $[0, T]$, the total sum of the time-delays must be less than T , i.e., $0 \leq \delta_1 + \dots + \delta_J \leq T$. In case of $J = 1$, we can easily obtain the following formula.

$$R(1; T) = \int_0^T r \exp(-r\delta_1) d\delta_1 = 1 - \exp(-rT). \tag{2}$$

In case of $J \geq 2$, due to the independence of time-delay trials, we can calculate the probability $R(J; T)$ as follows:

$$R(J; T) = \int_0^T \int_0^{T-\delta_1} \dots \int_0^{T-(\delta_1+\dots+\delta_{J-1})} \prod_{j=1}^J r \exp(-r\delta_j) d\delta_1 \dots d\delta_J \tag{3}$$

Here by noting the following two formulas,

$$\int_0^{T-(\delta_1+\dots+\delta_{J-1})} r \exp(-r\delta_J) d\delta_J = 1 - \exp(-rT) \prod_{j=1}^{J-1} \exp(r\delta_j),$$

$$\int_0^T \dots \int_0^{T-(\delta_1+\dots+\delta_{J-2})} r^{J-1} \exp(-rT) d\delta_1 \dots d\delta_{J-1} = \exp(-rT) \frac{(rT)^{J-1}}{(J-1)!},$$

we can calculate Eq. 3 as follows:

$$R(J; T) = R(J-1; T) - \exp(-rT) \frac{(rT)^{J-1}}{(J-1)!} \tag{4}$$

Therefore, from Eqs. 2 and 4, we can derive the following explicit formula:

$$R(J; T) = 1 - \exp(-rT) \sum_{j=1}^J \frac{(rT)^{j-1}}{(j-1)!}. \tag{5}$$

Here, we can easily see that $R(J; T)$ is a monotonic decreasing function approaching to zero as J increases.

Now, by combining Eqs. 1 and 5, we can derive a new method for estimating the cumulative influence degree within time-interval $[T_0, T_1]$ for arbitrary diffusion probabilities. We can formulate the key formula as follows:

$$\sigma_{[T_0, T_1]} = \sum_{J=1}^{\infty} R(J; T_0, T_1) P^J \mathbf{1}. \tag{6}$$

Below we can summarize the algorithm of the proposed method.

1. Set each element of $\sigma_{[T_0, T_1]}$ to 0, and set $J \leftarrow 1$ and $\mathbf{x} \leftarrow \mathbf{1}$.
2. Calculate $\mathbf{x} \leftarrow P\mathbf{x}$ and if $R(J; T_0, T_1)\|\mathbf{x}\| < \eta$, then output $\sigma_{[T_0, T_1]}$ and terminate.
3. Set $\sigma_{[T_0, T_1]} \leftarrow \sigma_{[T_0, T_1]} + R(J; T_0, T_1)\mathbf{x}$ and $J \leftarrow J + 1$ and return to 2.

In this algorithm, $\mathbf{x} \in \mathbb{R}^{|V|}$ is a vector to calculate the expected number of the J -step reachable nodes, and η is a parameter for the termination condition. In our experiments, η is set to a sufficiently small number, i.e., 10^{-12} .

4 Experiments

We first evaluate the performance (accuracy) of the proposed method (*iterative method*) by comparing with the *naive simulation method* with different number of runs to estimate the empirical mean using three large real social networks. We then compare the *iterative method* with two other methods, the *infinite iterative method* and the *SIS with fixed time delay method* in terms of the estimated *cumulative influence degree* for the CTSIS model using the same networks. Finally we compare the efficiency (computation time) of the *iterative method* with the *naive simulation method*. In all the experiments, we consider the simplest case where the both diffusion and time-delay parameters of the CTSIS model are uniform for any link.

4.1 Datasets

We employed three datasets of large real networks. These are all bidirectionally connected networks. The first one is a network of people that was derived from the “list of people” within Japanese Wikipedia, also used in [15], and has 9,481 nodes and 245,044 directed links (the Wikipedia network). The second one is a network derived from the Enron Email Dataset [16] by extracting the senders and the recipients and linking those that had bidirectional communications, and has 4,254 nodes and 44,314 directed links (the Enron network). The third one is a Coauthorship network used in [17] and has 12,357 nodes and 38,896 directed links (the coauthorship network).

4.2 Accuracy Evaluation

We evaluated the accuracy of the proposed method by comparing it with the *naive simulation method* mentioned in section 2.3. We speculate that the *cumulative influence degree* estimated by taking the empirical mean of the results of the *naive simulation method* converges asymptotically to the true value as the number of simulations M increases. Thus, we first examined how the difference of the estimated cumulative influence degree between the *iterative method* and the *naive simulation method* changes as M changes for the three networks.

The difference was evaluated by

$$\epsilon_M = \sum_{v \in V} |\sigma(v; T) - s_M(v; T)| / |V|, \quad (7)$$

where $\sigma(v; T)$ and $s_M(v; T)$ are the *cumulative influence degree* of node v estimated by the *iterative method* and the *naive simulation method*, respectively. We used $T = 10^4$ and varied M from 100, 1,000, and 10,000.

In these experiments we determined the values for the diffusion and time-delay parameters as follows. As noted in 3.1, it is required that the diffusion parameter κ must be less than $\text{eig}(\mathbf{A})^{-1}$, the reciprocal of the largest eigenvalue of the adjacency matrix \mathbf{A} of the network for the *infinite iterative method* to obtain a finite value of σ_∞ . The values of $\text{eig}(\mathbf{A})^{-1}$ for the Wikipedia, Enron, and Coauthorship networks were 0.00674, 0.0205, and 0.105, respectively. Thus, we adopted 0.0067, 0.02, and 0.1 as the values of κ for these networks, respectively. These are the largest values that the *infinite iterative method* can take. We set $r = 1$ for the time-delay parameter. This is equivalent to setting the average time delay to be a unit time which is consistent to the discrete time step of the *SIS with fixed time delay method*.

Table 1 summarizes the results, from which we can see that the estimation difference decreases as M increases and it becomes reasonably small at $M = 10,000$ for all the three networks. We are able to verify our speculation and conclude that the proposed *iterative method* can indeed estimate the *cumulative influence* accurately.

Table 1. Estimation difference between the *iterative method* (proposed) and the *naive simulation method*

network	M		
	100	1,000	10,000
Wikipedia	0.196	0.062	0.020
Enron	0.552	0.190	0.062
Coauthorship	0.298	0.096	0.031

4.3 Cumulative Influence Degree Comparison

Next, we investigated how well the other approaches can approximate the *cumulative influence degree*. We compared two approaches. One is the *infinite iterative method* described in 3.1. The other is the *SIS with fixed time delay method* [11]⁵. The *SIS with fixed time delay method* uses bond percolation on the layered graph which is constructed from the original social network with each layer added on top as the time proceeds[10] and much more efficiently estimates the *cumulative influence degree* than the *naive simulation method*. We used the same $M (= 10,000)$ from the result in 4.2. For each network, we investigated two cases, one with a short time span $T = 10$ and the other with a long time span $T = 100$. Note that we set $r=1$ and thus, the average time delay $\bar{\delta} = 1$. We selected the top 200 most influential nodes that the *iterative method* identified and compared their *cumulative influence degree* with the values that the other two methods estimated for the same 200 nodes.

Figure 1 illustrates the results of comparison. We can see that the *infinite iterative method* estimate the *cumulative influence degree* fairly well for a long time span $T = 100$ except for the Wikipedia network, but it tends to overestimate it for a short time span $T = 10$. In contrast, the *SIS with fixed time delay method* tends to underestimate

⁴ We had to set the value to be small so that the naive simulation returns the result within a day.

⁵ Note that in [11] the influence degree was defined to be the expected number of active nodes at the end of observation time T , but here the algorithm in [11] is modified to calculate the *cumulative influence degree*.

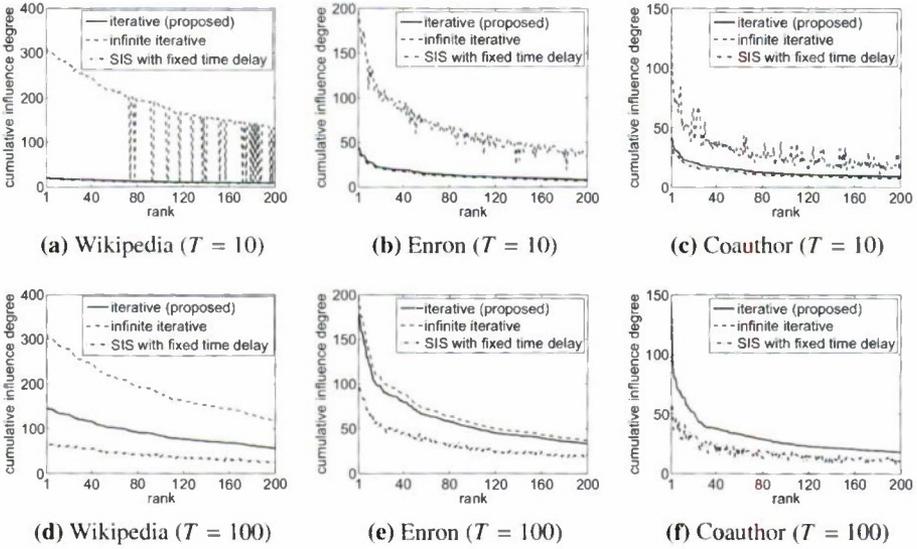


Fig. 1. Comparison in cumulative influence degrees of top 200 influential nodes

the *cumulative influence degree* for a large time span $T = 100$ but it does well for a short time span $T = 10$. These results show that these two methods cannot correctly estimate the *cumulative influence degree* for an arbitrary time span.

It is noted that there are many bumps in the graphs for the cases where the estimation of the other two methods is very poor, i.e. $T = 10$ for the *infinite iterative method* and $T = 100$ for the *SIS with fixed time delay method*. This implies that the ranking results by these methods are different from the true ranking by the *iterative method*. The curves becomes smoother when the estimation becomes better.

4.4 Efficiency Evaluation

We see in 4.3 that both *infinite iterative method* and *SIS with fixed time delay method* do not accurately estimate the *cumulative influence degree*, and we compare the computation time of the *iterative method* with the *naive simulation method* for $M = 1$. The results are shown in Fig. 2 for three values of the time span $T = 10, 20, 100$ and for each of the three networks. Three values are chosen for κ . The minimum values are the same as the ones used in 4.2 and 4.3, and the other values are obtained by multiplying 1.5 in sequence. The *iterative method* returns the values in less than 0.5 sec. for all cases and very insensitive to the parameter values. The *naive simulation method* is only efficient when the κ is very small and requires exponentially increasing time as κ increase. In deed it did not return the values within 3 days in many cases. Considering that this is for a single simulation, use of the *naive simulation method* is not practical and out of question.

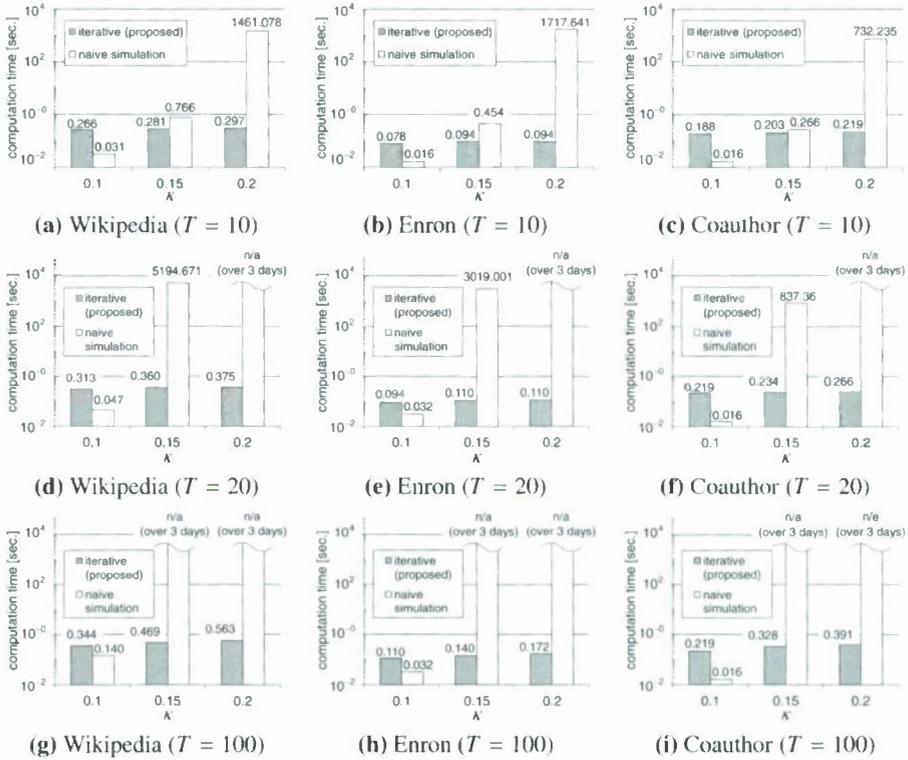


Fig. 2. Comparison in computation time

5 Discussions

We mentioned in 3.1 that the *cumulative influence degree* derived by the *infinite iterative method* is similar to the centrality proposed by Bonacich [13] and identical to the Katz' measure [14]. In [13] the standard centrality e_u of node u is defined by

$$\lambda e_u = \sum_{v \in V} a_{u,v} e_v, \quad (8)$$

where λ is a constant introduced to ensure a non-zero solution, and A is the adjacency matrix ($a_{u,v}$ is its element) as before. Bonacich generalized Eq. 8 by introducing the strength of relationship β , which is equivalent to κ in this paper, and derived the generalized centrality $c_u(\alpha, \beta)$ as

$$c_u(\alpha, \beta) = \sum_{v \in V} (\alpha + \beta c_v(\alpha, \beta)) a_{u,v}, \quad (9)$$

where α is a normalization constant. It is easily shown that $c_i(\alpha, \beta)$ is written in a matrix notation as

$$c(\alpha, \beta) = \alpha \sum_{J=0}^{\infty} \beta^J A^{J+1} \mathbf{1} = \alpha (A \mathbf{1} + \beta A^2 \mathbf{1} + \beta^2 A^3 \mathbf{1} + \dots). \quad (10)$$

Comparing Eq. 1 with Eq. 10, we note that they are the same except that the generalized centrality assumes that the strength of relationship with the directed connected nodes is 1. Further, we note that the following equality holds.

$$\sigma_{\infty} = \frac{\beta}{\alpha} c(\alpha, \beta), \quad (11)$$

which is exactly the same as Katz's measure. Thus, the *cumulative influence degree* σ_{∞} defined by the *infinite iterative method* is interpreted as a centrality measure.

We showed in 4.3 that the *infinite iterative method* well approximates the *cumulative influence degree* when the time span is large. This is evident because the *infinite iterative method* assumes an infinite time span. In the extreme limit of $T = \infty$, the *iterative method* converges to the infinite iterative method. How large T should be in order for it to be large depends on the delay time parameter r . When r gets smaller, a smaller T can be called large, e.g. $T = 10$ is large when $r = 0.1$. Similar argument can be made for the *SIS with fixed time delay method*. The *SIS with fixed time delay method* advances the time in a discrete step. Thus, it happens that multiple parents attempt to activate the same node simultaneously at the same time. If this happens, the activation count is only incremented by one. When the time span T is small, the diffusion propagation does not go far and there is not much chance that this simultaneous activation happens. This is why the *SIS with fixed time delay method* gives good results for a small time span T . However, how good the *SIS with fixed time delay method* approximates the *cumulative influence degree* depends on how close the time step is to the average delay-time $\bar{\delta}$. It overestimates the true *cumulative influence degree* for $T = 10$ when $r = 0.1$ and underestimates it when $r = 10$. We confirmed this by additional experiments but due to the space limit we do not show the figures.

6 Conclusion

In this paper we addressed the problem of efficiently estimating the *cumulative influence degree* of a node in social networks when the information diffusion follows the Susceptible/Infective/Susceptible (SIS) model with asynchronous continuous time delay based on the independent cascade framework. It is possible to analytically derive a formula by which to iteratively calculate the *cumulative influence degree* to a desired accuracy. The simplified version which corresponds to assuming an infinitely large time span is closely related to the generalized centrality measure. We showed by applying the method to three large real world social networks that the method can accurately estimate the *cumulative influence degree* with 2 to 6 orders of magnitude less computation time than the *naive simulation method*. Thus, it can be used to rank the influential nodes very efficiently. We also compared the proposed *iterative method* to the *SIS with fixed time delay model* and the *infinite iterative method* and confirmed that they generally produce poor estimates and only give good results when a specific condition holds for each.

Acknowledgments

This work was partly supported by Asian Office of Aerospace Research and Development, Air Force Office of Scientific Research under Grant No. AOARD-08-4027, and JSPS Grant-in-Aid for Scientific Research (C) (No. 20500147).

References

1. Newman, M.E.J., Forrest, S., Balthrop, J.: Email networks and the spread of computer viruses. *Physical Review E* 66, 035101 (2002)
2. Newman, M.E.J.: The structure and function of complex networks. *SIAM Review* 45, 167–256 (2003)
3. Gruhl, D., Guha, R., Liben-Nowell, D., Tomkins, A.: Information diffusion through blogspace. *SIGKDD Explorations* 6, 43–52 (2004)
4. Domingos, P.: Mining social networks for viral marketing. *IEEE Intelligent Systems* 20, 80–82 (2005)
5. Leskovec, J., Adamic, L.A., Huberman, B.A.: The dynamics of viral marketing. In: *Proceedings of the 7th ACM Conference on Electronic Commerce (EC 2006)*, pp. 228–237 (2006)
6. Goldenberg, J., Libai, B., Muller, E.: Talk of the network: A complex systems look at the underlying process of word-of-mouth. *Marketing Letters* 12, 211–223 (2001)
7. Kempe, D., Kleinberg, J., Tardos, E.: Maximizing the spread of influence through a social network. In: *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2003)*, pp. 137–146 (2003)
8. Kimura, M., Saito, K., Nakano, R.: Extracting influential nodes for information diffusion on a social network. In: *Proceedings of the 22nd AAAI Conference on Artificial Intelligence (AAAI 2007)*, pp. 1371–1376 (2007)
9. Kimura, M., Saito, K., Motoda, H.: Blocking links to minimize contamination spread in a social network. *ACM Transactions on Knowledge Discovery from Data* 3, 9:1–9:23 (2009)
10. Kimura, M., Saito, K., Motoda, H.: Efficient estimation of influence functions for SIS model on social networks. In: *Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI 2009)*, pp. 2046–2051 (2009)
11. Saito, K., Kimura, M., Motoda, H.: Discovering influential nodes for SIS models in social networks. In: Gama, J., Costa, V.S., Jorge, A.M., Brazdil, P.B. (eds.) *DS 2009*. LNCS (LNAI), vol. 5808, pp. 302–316. Springer, Heidelberg (2009)
12. Saito, K., Kimura, M., Ohara, K., Motoda, H.: Learning continuous-time information diffusion model for social behavioral data analysis. In: Zhou, Z.-H., Washio, T. (eds.) *ACML 2009*. LNCS (LNAI), vol. 5828, pp. 322–337. Springer, Heidelberg (2009)
13. Bonacichi, P.: Power and centrality: A family of measures. *American Journal of Sociology* 92, 1170–1182 (1987)
14. Katz, L.: A new status index derived from sociometric analysis. *Sociometry* 18, 39–43 (1953)
15. Kimura, M., Saito, K., Motoda, H.: Minimizing the spread of contamination by blocking links in a network. In: *Proceedings of the 23rd AAAI Conference on Artificial Intelligence (AAAI 2008)*, pp. 1175–1180 (2008)
16. Klimt, B., Yang, Y.: The enron corpus: A new dataset for email classification research. In: Boulicaut, J.-F., Esposito, F., Giannotti, F., Pedreschi, D. (eds.) *ECML 2004*. LNCS (LNAI), vol. 3201, pp. 217–226. Springer, Heidelberg (2004)
17. Palla, G., Derényi, I., Farkas, I., Vicsek, T.: Uncovering the overlapping community structure of complex networks in nature and society. *Nature* 435, 814–818 (2005)

Two Natural Heuristics for 3D Packing with Practical Loading Constraints*

Lei Wang¹, Songshan Guo¹, Shi Chen¹, Wenbin Zhu^{2,3,**}, and Andrew Lim⁴

¹ Department of Computer Science, School of Information Science and Technology, Zhong Shan (Sun Yat-Sen) University, Guangzhou, Guangdong, P.R. China (510275)
is03wlei@gmail.com, issgssh@mail.sysu.edu.cn, iamcs1983@gmail.com

² Department of Computer Science and Engineering, School of Engineering, Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong S.A.R.
i@zhuwb.com

³ Department of Logistics and Maritime Studies, Faculty of Business, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong S.A.R.

⁴ Department of Management Sciences, College of Business, City University of Hong Kong, Tat Chee Ave, Kowloon Tong, Hong Kong S.A.R.
lim.andrew@cityu.edu.hk

Abstract. In this paper, we describe two heuristics for the Single Vehicle Loading Problem (SVLP), which can handle practical constraints that are frequently encountered in the freight transportation industry, such as the servicing order of clients; item fragility; and the stability of the goods. The two heuristics, Deepest-Bottom-Left-Fill and Maximum Touching Area, are 3D extensions of natural heuristics that have previously only been applied to 2D packing problems. We employ these heuristics as part of a two-phase tabu search algorithm for the Three-Dimensional Loading Capacitated Vehicle Routing Problem (3L-CVRP), where the task is to serve all customers using a homogeneous fleet of vehicles at minimum traveling cost. The resultant algorithm produces mostly superior solutions to existing approaches, and appears to scale better with problem size.

Keywords: vehicle routing, 3D packing, Deepest-Bottom-Left-Fill, Maximum Touching Area, Tabu Search.

1 Introduction

The Three-Dimensional Loading Capacitated Vehicle Routing Problem (3L-CVRP) was first introduced by [1] and subsequently studied by [2]. The task is to plan the routes for a fleet of homogeneous vehicles that delivers items to customers, such that the total distance traveled by all vehicles is minimized. In addition, the three-dimensional loading plan for each vehicle must be formulated

* This research is partially supported by Niche Area Grant J-BB7C of the Hong Kong Polytechnic University.

** Corresponding author.

that fulfills constraints addressing issues such as the stability and fragility of the items and the convenience of loading and unloading.

When searching for a solution for 3L-CVRP, the Single Vehicle Loading (sub-) Problem (SVLP) must be solved multiple times. Our primary contribution in this paper is the introduction of two heuristics for the SVLP, namely the Deepest-Bottom-Left-Fill (DBLF) and the Maximum Touching Area (MTA) heuristics. These heuristics are used in a two-phase tabu search algorithm; the first phase attempts to make an infeasible initial solution feasible, and the second phase attempts to improve the quality of the solution.

We compared our DBLF+MTA tabu search (DMTS) algorithm with the Tabu Search (TS) algorithm employed by [1] and the Ant Colony Optimization (ACO) algorithm designed by [2] using a standard set of 27 test cases. Our experiments show that DMTS outperforms TS in all cases, and produces a superior solution to ACO for 22 out of the 27 cases. Furthermore, the running time for DMTS does not dramatically increase with problem size unlike TS and ACO, and it converges to good solutions rapidly.

2 Related Work

The many variants of the Capacitated Vehicle Routing Problem (CVRP) have two common aspects, namely *routing* and *loading*. The goal of routing is to determine a sequence that visits all customers with minimum total travel cost. The goal of loading is to find a loading plan for each vehicle that satisfies all loading constraints. Instead of considering the actual packing of each item, a scalar value is usually used to represent the volume of each item; as long as the total volume of the items loaded does not exceed the vehicle's capacity, it is assumed that loading is possible [3].

[1] was the first to consider the vehicle routing problem with three-dimensional loading constraints (3L-CVRP). A tabu search approach was proposed to address the 3L-CVRP, where the three-dimensional loading sub-problem was also solved by a tabu search metaheuristic. [2] employed a local search to solve the loading sub-problem along with an ant colony optimization routine to find an overall solution to the 3L-CVRP problem.

There are approaches for the 2D loading problem that can potentially be adapted to 3L-CVRP, for example [4,5], but it is unclear how such approaches can handle practical constraints.

The SVLP is related to container loading problems. Various practical constraints on the supporting surface or item fragility are often considered [6,7,8]. To date, the best results on problems of reasonable size are held by heuristic-based methods [9,10,11,12].

3 Problem Description

Let $G = (V, E)$ be an undirected graph, where $V = \{0, 1, 2, \dots, n\}$ is the set of $n + 1$ vertices corresponding to a depot, represented by vertex 0, and n clients,

denoted by vertices $1, \dots, n$; and E is the set of edges. The cost of an edge (i, j) is denoted by c_{ij} . There are v identical vehicles available; each vehicle has a weight capacity D and a three-dimensional rectangular loading space $S = W \times H \times L$ defined by width W ; height H ; and length L . Each client $i (i = 1, \dots, n)$ requires the delivery of a set of m_i three-dimensional items $I_{ik} (k = 1, 2, \dots, m_i)$ having width w_{ik} , height h_{ik} and length l_{ik} with total weight d_i .

In 3L-CVRP, we assume all items are rectangular boxes. The items can only be placed orthogonally inside a vehicle; however, items can be rotated by 90° on the width-length plane. Some items are also marked *fragile*.

The objective of 3L-CVRP is to find a set of at most v routes (one per vehicle) such that

- (1) Every vehicle starts from the depot, visits a sequence of clients and returns to the depot;
- (2) All clients are served by exactly one vehicle;
- (3) No vehicle carries a total weight that exceeds its capacity;
- (4) All items for a particular vehicle can be orthogonally packed while satisfying the following *loading constraints*:
 - (4.a) (*Fragility Constraint*) no non-fragile items are placed on top of fragile items;
 - (4.b) (*Supporting Area Constraint*) all items have a supporting area of at least α percent of their base area;
 - (4.c) (*LIFO constraint*) all items fulfill the LIFO policy, i.e., when client i is visited, all of its corresponding items I_{ik} must not be stacked beneath nor be blocked by items of later clients. An item is considered blocked if it will overlap any item of a later client when it is moved along the L axis towards the door.
- (5) The total cost of all edges in the routes is minimized

In this study, we use the following Cartesian coordinate system. The loading space of the vehicle is in the first octant and the origin is at the deepest, bottommost, leftmost corner. The width W ; the height H ; and the length L is parallel to the x -; y -; and z -axis respectively. The terms *left*; *right*; *top*; *bottom*; *back*; and *front* are self-explanatory. The vehicle has a single door, which is located at the front (i.e., at $z = L$).

4 Procedure for the SVLP

In order to solve the 3L-CVRP, we must solve the SVLP, defined as follows. Given a list of clients to be visited in a fixed order, devise a loading plan for all items for a particular vehicle that satisfies all the loading constraints. We can do this by treating the vehicle as an open-ended bin (i.e., vehicle of infinite length), try to find the minimum length to accommodate all the items, and then compare this value with the length of the vehicle.

Our SVLP procedure makes use of a routine *LoadAll*. Given an ordered list of items L , *LoadAll* returns the minimum length required to load all items into

Algorithm 1. Local Search for the SVLP

```

input :  $L$ : list of items
1 if total volume or weight of items in  $L$  exceeds capacity of vehicle then return
  failure;
2 Sort  $L$  by reverse visiting order, then non-fragile first, then by descending
  volume;
3 for  $k = 1$  to  $K$  do
4    $len \leftarrow \text{LoadAll}(L)$ ;
5   if  $len \leq \text{length of vehicle}$  then return success;
6   Randomly swap items  $i$  and  $j$  in  $L$ ,  $i \neq j$ ;
7 return failure;

```

the vehicle. Assuming the existence of *LoadAll*, we can use Algorithm 1 to solve the SVLP (we set $K = 150$ in our experiments).

The *LoadAll* procedure uses two heuristics, namely Deepest-Bottom-Left-Fill (DBLF) and Maximum Touching Area (MTA). Both heuristics attempt to produce a plan that minimizes the length required to load all items; the *LoadAll* routine invokes both heuristics and returns the smaller result.

5 The DBLF Heuristic

The Deepest-Bottom-Left-Fill (DBLF) heuristic loads items one at a time by placing the current item in the deepest, bottommost, left-most position. It is an adaptation of the Bottom-Left-Fill (BLF) heuristic for 2D packing [13] into three dimensions.

For any packing pattern, there is an equivalent pattern where each item is pushed as close to the origin as possible called a *normal pattern*. Correspondingly, given an existing packing pattern, all positions that an item can occupy following this rule are called *normal positions*. It is therefore sufficient when following the DBLF rule to only try normal positions when placing an item, i.e., the positions where the back face of item i touches either the front face of some already packed items or the vehicle; the left side of item i touches either the right side of some already packed items or the vehicle; and the bottom of item i touches either the top of some already packed items or the vehicle.

In this section, we present two versions of the DBLF heuristic. The first version does not include the supporting area constraint, but does handle the other constraints (fragility and LIFO); the second version contains the modifications necessary to handle the supporting area constraint. Both versions run in $O(n^4)$ time, where n is the number of items per vehicle.

5.1 DBLF without Supporting Area Constraint

An item i is said to be *placed at* x_i (similarly for y_i or z_i) if x_i is the x -coordinate of the left (or bottom or back) face of an item i . Let w_i (respectively h_i or l_i)

be the magnitude of the projection of item i onto the x -axis; the x -coordinate of the right (or top or front) face is then given by $x_i + w_i$.

Assuming $i - 1$ items have been loaded, consider the i -th item. If y_i and z_i are fixed, then enumerating all candidates for x_i and checking the feasibility of candidate positions (x_i, y_i, z_i) for non-overlapping can be done in a single pass by sliding the item from left to right (see Algorithm 2). To do so, we maintain three lists of loaded items $L_{left}, L_{top}, L_{back}$ sorted in ascending order, where L_{left} is sorted by x -coordinate of the left face; L_{top} is sorted by y -coordinate of the top face; and L_{back} is sorted by z -coordinate of the back face. Note that in line 2 we introduced two dummy items *Bottom* and *Back* of dimensions $L \times W \times 0$ and $0 \times W \times H$ respectively; we initialize L_{top} and L_{back} to contain *Bottom* and *Back* respectively. This avoids having to check the boundary conditions of the vehicle itself.

Algorithm 2. DBLF without Supporting Area

```

input: List: list of  $m$  items
1 Initialize  $L_{left}$  with no items;
2 Initialize  $L_{top}, L_{back}$  with Bottom and Back;
3 for every item  $i$  in List do
4   best position  $pos \leftarrow (\infty, \infty, \infty)$ ;
5   best orientation  $(w, h, l) \leftarrow (\infty, \infty, \infty)$ ;
6   for each orientation  $w_i, h_i, l_i$  of item  $i$  do
7     for each item  $j$  in  $L_{back}$  do
8        $z_i \leftarrow z_j + l_j$ ;
9       for each item  $k$  in  $L_{top}$  s.t.  $y_k + h_k + h_i \leq H$  do
10         $y_i \leftarrow y_k + h_k$ ;
11         $p \leftarrow 0, x_i \leftarrow 0$ ;
12        while  $p < \text{size of } L_{left}$  and  $x_i + w_i \leq W$  do
13           $q \leftarrow$  be  $p$ -th item in  $L_{left}$ ;
14          if  $x_i + w_i \leq x_q$  then found  $x_i$  and break;
15          if item  $i$  overlaps with item  $q$  then  $x_i \leftarrow x_q + w_q$ ;
16           $p \leftarrow p + 1$ ;
17        if  $x_i + w_i \leq W$  then
18          find feasible position  $(x_i, y_i, z_i)$ ;
19          update  $pos$  and  $(w, h, l)$  if  $(x_i, y_i, z_i)$  is better;
20          continue next orientation (line 2);
21   place item  $i$  at  $pos$ ;
22   insert item  $i$  into  $L_{left}, L_{top}, L_{back}$ ;
23 return the largest  $z_i + l_i$ ;

```

Both the fragility and LIFO constraints can be handled without any increase in the time complexity by amending line 2 so that it also checks whether placing item i at x_i violates the fragility and LIFO constraints with respect to item q in constant time. Algorithm 2 runs in $O(n^4)$ time.

5.2 DBLF with Supporting Area Constraint

With the supporting area constraint, it is not sufficient to try only normal positions to honour the DBL rule. Consider a normal position that is feasible except that the supporting area is just below $\alpha\%$. If we move the item slightly to the right or the front before reaching the next normal position, the supporting area may now be sufficient.

Assume that the first $i - 1$ items have been loaded. Let $A(x, y, z)$ be the total supporting area contributed by the loaded items to item i when it is placed at (x, y, z) .

We first consider the case where y and z are fixed. Let $A_q(x, y, z)$ be the supporting area contributed by item q . Let B_i be the bottom face of item i ; T_q be the top face of item q ; and intervals I_{zi} and I_{zq} be the projections of B_i and T_q on the z -axis respectively. Obviously, $A_q(x, y, z) = 0$ if $y \neq y_q + h_q$ (i.e., the bottom of i is not at the same level as the top of q) or $I_{zi} \cap I_{zq} = \emptyset$. Otherwise, let s_{iq} be a line segment on the z -axis corresponding to $I_{zi} \cap I_{zq}$.

We can decompose $A_q(x, y, z)$ into four parts:

- (1) $A_{q1}(x, y, z)$ is the area swept by s_{iq} from the left side of T_q to the right side of B_i if $x_q < x + w_i$; 0 otherwise.
- (2) $A_{q2}(x, y, z)$ is the area swept by s_{iq} from the left side of T_q to the left side of B_i if $x_q < x$; 0 otherwise.
- (3) $A_{q3}(x, y, z)$ is the area swept by s_{iq} from the right side of T_q to the right side of B_i if $x_q + w_q < x + w_i$; 0 otherwise
- (4) $A_{q4}(x, y, z)$ is the area swept by s_{iq} from the right side of T_q to the left side of B_i if $x_q + w_q < x$; 0 otherwise.

Observe that $A_q(x, y, z) = A_{q1}(x, y, z) - A_{q2}(x, y, z) - A_{q3}(x, y, z) + A_{q4}(x, y, z)$ for any position of i and q . Let $A_r(x, y, z)$, $r = 1, 2, 3, 4$ be the sum of $A_{qr}(x_i, y_i, z_i)$ over all items q ; then $A(x, y, z) = A_1(x, y, z) - A_2(x, y, z) - A_3(x, y, z) + A_4(x, y, z)$. This is a useful observation because $A_r(x, y, z)$ can be easily computed.

For item q , we call $x_{q1}^* = x_q - w_i$; $x_{q2}^* = x_q$; $x_{q3}^* = x_q + w_q - w_i$; and $x_{q4}^* = x_q + w_q$ the *event points* of A_1, A_2, A_3 and A_4 respectively. This is because when item i slides from left to right, after the event point of A_r , item q starts to contribute to A_r .

Let X_r be the set of all event points of A_r sorted in ascending order. $A_r(x, y, z)$ is a piecewise linear function of x with local maxima achieved at $x \in X_r$. Note that the slope between two consecutive event points is constant. Let $SZ_r(x, y, z)$ be the slope of A_r at (x, y, z) ; since the contribution by item q to A_r changes only at the event points, $SZ_r(x, y, z)$ is a step function of x .

Let X be the set of all event points in $X_r, r = 1, 2, 3, 4$ sorted in ascending order. $A(x, y, z)$ is a piecewise linear function of x with maxima achieved at some $x \in X$.

When item i slides along the z -axis with x, y fixed, we can also define the event points Z_r for A_r and the slope function $SX_r(x, y, z)$. Using similar arguments, we find that $A_r(x, y, z)$ is a piecewise linear function of z with local maxima achieved at $z \in Z_r$; $SX_r(x, y, z)$ is a step function of z ; and if Z is the set of all

event points in Z_r sorted in ascending order, then $A(x, y, z)$ is a piecewise linear function with local maxima at $z \in Z$. Hence, Lemma 1 holds:

Lemma 1. *When y is fixed, the local maxima of $A(x, y, z)$ is at some $(x, z) \in X \times Z$, where X and Z are event points.*

Once the values of $A_r(x, y, z)$ and $Z_r(x, y, z)$ at all pairs (x, z) in $X \times Z$ are known, then $A(x, y, z)$ can be computed.

Let $NO(x, y, z)$; $F(x, y, z)$; and $LIFO(x, y, z)$ be indicator functions where 1 indicates that placing an item at (x, y, z) will satisfy the non-overlapping; fragility; and LIFO constraint with respect to all other items, respectively, and 0 otherwise. These three functions can be efficiently computed; we will use $NO(x, y, z)$ for illustration.

We can divide the base of the vehicle into $|X| \times |Z|$ grid squares using lines parallel to the z - and x -axis that pass through points in X and Z respectively. The following lemma is readily verified.

Lemma 2. *$NO(x_1, y, z_1) = NO(x_2, y, z_2)$ if (x_1, z_1) and (x_2, z_2) are in the interior of the same grid. The same is true for $F(x, y, z)$ and $LIFO(x, y, z)$.*

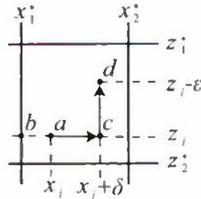


Fig. 1. Illustration of proof of Theorem 1

Theorem 1. *If (x_i, y_i, z_i) is the best DBL position for item i that satisfies the non-overlapping; fragility; and LIFO constraints, then either $x_i \in X$ or $z_i \in Z$.*

Proof. Assume on the contrary, i.e., the best position $a = (x_i, y_i, z_i)$ is inside a grid square (Figure 1). By Lemma 2, any position (in particular b, c, d) in the same square will also satisfy the non-overlapping; fragility; and LIFO constraints.

Suppose we slide item i leftwards along the line $z = z_i$. The supporting area $A(x, y_i, z_i)$ is a linear function of x . If the slope is not positive, then position b will have a larger or the same supporting area as a . Since b is feasible, this contradicts the fact that (x_i, y_i, z_i) is the leftmost position; hence, the slope must be positive.

Now suppose we slide item i along the line $z = z_i$ by δ towards the right to position c . Since the slope is positive, the supporting area at position c is strictly greater than a . At this point, we can slide the item along $x = x_i + \delta$ by ϵ . Since $A(x_i + \delta, y_i, z)$ is a linear function of z , we can find a position d with supporting area greater than or equal to at a . Since position d is deeper than (x_i, y_i, z_i) , a does not respect the DBL rule. Hence, either $x_i \in X$, or $z_i \in Z$.

Let $cap(x, X) = \min\{r \geq x : r \in X\}$ be a function that returns the smallest number in X that is greater than x . Using a similar argument as Lemma 2:

Lemma 3. *If (x, y, z) is the best DBL position for item i and satisfies all loading constraints, then $(cap(x, X), y, cap(z, Z))$ is a feasible position.*

Theorem 1 and Lemma 3 implies that there exists a best position for item i , (x_i, y, z_i) , that is either the result of pushing item i from location (x, y, z) along the z -axis to its deepest position or along the x -axis to its leftmost position in the grid square. Consequently, we only need to search for feasible positions $(x, y, z), \forall (x, z) \in X \times Z$.

In order to find the best DBL position, we first scan all possible $z \in Z$, then $y \in Y$ in ascending order. For each z, y we examine all $x \in X$. If a feasible position is found with supporting area larger than $\alpha\%$, then it is possible that item i can be pushed deeper and/or to the left. Let $cup(z, Z)$ be the largest element smaller than z in Z ; we can push the item deeper only if at the position $(x, y, cup(z, Z))$, all indicators NO ; F ; and $LIFO$ are 1, or to the left if all indicators are 1 at the position $(cup(x, X), y, z)$.

The values of X ; Z ; all indicator functions; and the supporting area can be computed in linear time. Pushing item i deeper or to the left can be done in constant time (since $A(x, y, z)$ is a linear function of z and x). The time complexity to load a single item is therefore $O(n^3)$; hence, to load all n items, the total time complexity is $O(n^4)$.

6 Maximum Touching Area Heuristic

The Maximum Touching Area (MTA) heuristic places items into the vehicle at the position that maximizes the total contact area of its faces with the faces of other items or with the vehicle. It is an extension of the Maximum Touching Perimeter heuristic for 2D packing [14] into three dimensions.

Let $A_{left}(x, y, z)$; $A_{right}(x, y, z)$; $A_{front}(x, y, z)$; $A_{back}(x, y, z)$; $A_{top}(x, y, z)$; and $A_{bottom}(x, y, z)$ be the contact area of the left; right; front; back; top; and bottom faces if the current item is placed at (x, y, z) , respectively. The total touching area $A_*(x, y, z)$ of the current item placed at (x, y, z) is the sum of these six functions.

Theorem 2. *Given a current loading pattern, there exists a position (x, y, z) , $x \in X, y \in Y, z \in Z$ to place the current item such that $A_*(x, y, z)$ is maximal.*

Proof. Consider $A_*(x, y, z)$ for arbitrary fixed y, z . Observe that $A_{front}(x, y, z)$; $A_{back}(x, y, z)$; $A_{top}(x, y, z)$; and $A_{bottom}(x, y, z)$ are all piecewise linear functions of x with extreme points at $x \in X$. The sum of these four functions is also a piecewise linear function of x with extreme points at $x \in X$. Furthermore, $A_{left}(x, y, z)$ and $A_{right}(x, y, z)$ can only be non-zero if $x \in X$. Therefore, the local maxima for $A_*(x, y, z)$ with arbitrary fixed y, z must be when $x \in X$.

Without loss of generality, the same applies when considering $A_*(x, y, z)$ for arbitrary fixed x, y or arbitrary fixed x, z . Therefore, the global maximal for $A_*(x, y, z)$ must be at some position (x, y, z) where $x \in X, y \in Y, z \in Z$.

We can adapt the DBLF algorithm to find the position that maximizes total contact area. Aside from the packing rule, there are two differences between MTA and DBLF: 1) for MTA we must scan all $(x, y, z) \in X \times Y \times Z$, whereas in DBLF we can stop as soon as a feasible position has been found; and 2) for MTA we need not push the current item along the grid lines unlike for DBLF.

7 A Two-Phase Tabu Search for the 3L-CVRP

We adapted the savings algorithm for CVRP [15] to construct our initial solution. Starting with one client per route, we iteratively merge two routes with the largest traveling time savings until no more merging can be done. If after the first round of merging there are more routes than vehicles, then we perform another round of merging, where we allow both the total weight of items and the length of loading space to exceed vehicle capacity; the second round ends when the number of routes and vehicles are equal.

We employed five neighbourhood operators, namely:

- **2-opt**: select a pair of clients (i, j) from a route; the order of all clients between i and j inclusive are reversed.
- **2-swap**: select a pair of clients from a route with at least 3 clients; the order of the selected pair is swapped.
- **move**: select a client from route R_i and an insertion point from route R_j , $j \neq i$; the client is deleted from R_i and inserted into R_j at the insertion point.
- **crossover**: select a splitting point from each of two routes R_i and R_j (with at least two clients); the prefix sequences of the two routes are exchanged.
- **splitting**: select a splitting point from a route R_i (with at least two clients); R_i is split into two routes at that point.

The five neighbourhood operators 2-opt; 2-swap; move; crossover; and splitting are assigned a hand-tuned weight of 1000; 1000; 3000; 4500; and 500 respectively, which provides the relative probability that the operator will be applied. We set the tabu tenure T to 30 for both phases. These values were determined after some preliminary investigation.

Phase one uses only the 2-swap, move and crossover operators, and is invoked only if the initial solution is infeasible. The following objective function obj to captures the excess weight and length:

$$obj = \text{route len.} + \alpha \cdot \text{excess wt.} + \beta \cdot \text{excess len.} \quad (1)$$

$$\alpha = 20\bar{c}/D \quad (2)$$

$$\beta = 20\bar{c}/L \quad (3)$$

where \bar{c} is the average cost of the edges; D is the capacity of a vehicle; and L is the length of a vehicle. The values of α and β are increased if no progress is made in 10 consecutive iterations; α will be increased by 50% if some route exceeds the weight capacity D , while β will be increased by 50% if some route requires a vehicle with length greater than L . We sample 500 neighbours in each iteration,

and the best solution is selected as the current solution for next iteration. Phase one continues until a feasible solution is found, or 10,000 iterations are reached.

Phase two uses all five operators, and only feasible solutions are generated. In each iteration, 500 neighbours are generated, and the one with minimum total traveling cost is selected as the current solution for the next iteration. The best solution found after 10,000 iterations is retained.

8 Computational Experiments

Our DBLF+MTA tabu search (DMTS) algorithm was coded in C++ using the g++ compiler. It was tested on a Hewlett-Packard server with an Intel Xeon E5430 2.66 GHz CPU, 8 GB RAM and running Linx (CentOS 5.1) using the 27 instances proposed by [1]. They can be broadly divided into three categories: *small* instances 1 to 9 have 15-25 customers with 32-50 items; *medium* instances

Table 1. Performance of TS vs ACO vs DMTS

No	TS		ACO		DMTS				Impr
	z	time(s)	avg	time(s)	min	avg	max	time(s)	
1	316.32	129.50	305.35	11.2	301.71	301.77	302.02	193.05	-1.17%
2	350.58	5.30	331.96	0.1	334.96	334.96	331.96	9.32	0.00%
3	447.73	461.10	409.79	88.5	387.34	387.91	387.97	87.05	-5.34%
4	448.48	181.10	440.68	3.9	437.19	438.59	440.68	91.30	-0.47%
5	464.24	75.80	453.19	22.7	436.48	440.23	445.69	444.72	-2.86%
6	504.46	1167.90	501.47	17.5	498.32	499.48	501.05	125.98	-0.40%
7	831.66	181.10	797.47	51.4	767.46	771.09	772.87	394.90	-3.31%
8	871.77	156.10	820.67	56.2	803.98	805.95	807.75	331.02	-1.79%
9	666.10	1468.50	635.50	15.3	630.13	630.90	634.00	197.90	-0.72%
10	911.16	714.00	841.12	241.2	826.39	832.46	836.52	707.07	-1.03%
11	819.36	396.40	821.04	172.4	768.25	781.85	788.60	820.90	-4.58%
12	651.58	268.10	629.07	46.2	610.23	614.78	619.43	194.76	-2.27%
13	2928.34	1639.10	2739.80	235.4	2697.70	2715.82	2725.97	859.47	-0.88%
14	1559.64	3451.60	1472.26	623.8	1428.99	1456.13	1483.45	1638.78	-1.10%
15	1452.34	2327.40	1405.48	621.0	1352.94	1371.26	1382.08	1537.39	-2.43%
16	707.85	2550.30	698.92	12.8	698.61	699.54	703.35	46.55	0.09%
17	920.87	2142.50	870.33	11.8	871.63	875.19	877.72	731.74	0.56%
18	1400.52	1452.90	1261.07	2122.2	1227.07	1248.28	1276.74	1748.84	-1.01%
19	871.29	1822.30	781.29	614.3	762.47	776.35	795.72	1376.97	-0.63%
20	732.12	790.00	611.26	3762.3	583.45	593.17	606.28	1647.83	-2.96%
21	1275.20	2370.30	1124.55	5140.0	1094.78	1121.60	1150.11	1594.57	-0.26%
22	1277.94	1611.30	1197.43	2233.6	1170.89	1176.76	1194.43	1287.74	-1.73%
23	1258.16	6725.60	1171.77	3693.4	1137.90	1148.02	1161.95	1091.05	-2.03%
24	1307.09	6619.30	1148.70	1762.8	1132.05	1144.56	1157.18	469.80	-0.36%
25	1570.72	5630.90	1436.32	8619.7	1434.00	1457.09	1469.05	1582.82	1.45%
26	1847.95	4123.70	1616.99	6651.2	1606.85	1616.61	1632.61	1488.72	-0.21%
27	1747.52	7127.20	1573.50	10325.8	1551.68	1574.23	1600.80	1440.18	0.05%
Avg	1042.26	2058.86	966.66	1746.6	946.43	956.10	966.26	820.02	-1.31%

10 to 18 have 29-44 customers with 62-94 items; and *large* instances 19 to 27 have 50-100 customers with 99-198 items.

We compared DMTS against the reported results of tabu search (TS) by [1] and the ant colony optimization (ACO) by [2]. The results for TS were obtained on a Pentium IV 3 GHz PC with 512MB RAM running Windows XP, and the results for ACO were obtained on a Pentium IV 3.2GHz with 2GB RAM running Linux. The CPU time limit for these two approaches was set to 1800 seconds for small instances 1-9; 3600 seconds for medium instances 10-18; and 7200 seconds for large instances 19-27. Since TS is deterministic, it was invoked once for each instance. Both ACO and DMTS were invoked 10 times with different random seeds on each run, and the average performance is reported.

The results are given in Table 1. The *time(s)* columns report the time taken to produce the best solution for each algorithm. Column *z* is the cost of the best solution found by TS. For ACO, the average total traveling cost over 10 executions is reported. We also report the minimum and maximum values for the total traveling cost for DMTS; the last column *Impr* gives the percentage improvement of the average of DMTS over the better result of TS and ACO; negative values indicate an improvement since the objective of this problem is to minimize the overall traveling cost.

Our experiments show that DMTS outperforms both TS and ACO in all instances except 2; 16; 17; 25; and 27. The average improvement is about 1.31%. We also see that the running times for TS and ACO do not scale well, increasing dramatically as the size of the instances increases. In contrast, DMTS maintains a similar running time for both the large and medium instances. Furthermore, the convergence rate for DMTS is very quick; in the majority of cases, a high quality solution can be found within the first 1000 iterations.

9 Conclusions

In this study, we examined the Three-Dimensional Loading Capacitated Vehicle Routing Problem. We extended two heuristics for the loading sub-problem, namely Deepest-Bottom-Left-Fill and Maximum Touching Area; for the overall algorithm, we employed a two-phase tabu search. Experiments showed that our DMTS algorithm outperforms the best known algorithms in 22 out of 27 instances and is significantly faster for large cases. Furthermore, the algorithm converges very quickly, so high quality solutions are discovered even in the very early stages of the search.

The DBLF and MTA heuristics are natural and logical ways to solve SVLP. They are sequence-based approaches that mimic the loading of a vehicle, and can effectively address the various constraints as each item is loaded. Although DMTS can be refined to produce better solutions (e.g., by introducing a branch and bound post-optimization step), it was primarily used to demonstrate the effectiveness of these heuristics, which can potentially be adapted to any problem that involves the SVLP with practical constraints.

References

1. Gendreau, M., Iori, M., Laporte, G., Martello, S.: A tabu search algorithm for a routing and container loading problem. *Transportation Science* 40(3), 342–350 (2006)
2. Fuellerer, G., Doerner, K.F., Hartl, R.F., Iori, M.: Metaheuristics for vehicle routing problems with three-dimensional loading constraints. *European Journal of Operational Research* 201(3), 751–759 (2010)
3. Toth, P., Vigo, D. (eds.): *The Vehicle Routing Problem. Monographs on Discrete Mathematics and Applications*. SIAM, Philadelphia (2001)
4. Chazelle: The bottom-left bin-packing heuristic: An efficient implementation. *IEEE Transactions on Computers* C-32(8), 697–707 (1983)
5. Healy, P.: An optimal algorithm for rectangle placement. *Operations Research Letters* 24(1-2), 73–80 (1999)
6. Bortfeldt, A.: A hybrid genetic algorithm for the container loading problem. *European Journal of Operational Research* 131(1), 143–161 (2001)
7. Eley, M.: Solving container loading problems by block arrangement. *European Journal of Operational Research* 141(2), 393–409 (2002)
8. Pisinger, D.: Heuristics for the container loading problem. *European Journal of Operational Research* 141(2), 382–392 (2002)
9. Lim, A., Zhang, X.: The container loading problem. In: *SAC 2005: Proceedings of the 2005 ACM Symposium on Applied Computing*, pp. 913–917. ACM, New York (2005)
10. Lim, A., Rodrigues, B., Yang, Y.: 3-d container packing heuristics. *Applied Intelligence* 22(2), 125–134 (2005)
11. Parreno, F., Alvarez-Valdes, R., Tamarit, J.M., Oliveira, J.F.: A maximal-space algorithm for the container loading problem. *Inform Journal on Computing* 20(3), 412–422 (2008)
12. Fanslan, T., Bortfeldt, A.: A tree search algorithm for solving the container loading problem. *Inform Journal on Computing*, ijoc.1090.0338+ (July 2009)
13. Baker, B.S., Coffman, E.G., Rivest, R.L.: Orthogonal packings in two dimensions. *SIAM Journal on Computing* 9(4), 846–855 (1980)
14. Lodi, A., Martello, S., Vigo, D.: Heuristic and metaheuristic approaches for a class of two-dimensional bin packing problems. *Inform Journal on Computing* 11(4), 345–357 (1999)
15. Clarke, G., Wright, J.W.: Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research* 12(4), 568–581 (1964)

Geometric Median-Shift over Riemannian Manifolds

Yang Wang and Xiaodi Huang

School of Computer Science and Technology, Tianjin University,
Tianjin, 300072 China

School of Computing and Mathematics, Charles Sturt University, Albury,
NSW 2640, Australia

wayag2000@yahoo.com.cn, xhuang@csu.edu.au

Abstract. Manifold clustering finds wide applications in many areas. In this paper, we propose a new kernel function that makes use of Riemannian geodesic distances among data points, and present a Geometric median shift algorithm over Riemannian Manifolds. Relying on the geometric median shift, together with geodesic distances, our approach is able to effectively cluster data points distributed over Riemannian manifolds. In addition to improving the clustering results, the complexity for calculating geometric median is reduced to $O(n^2)$, compared to $O(n^2 \log n^2)$ for Tukey median. Using both Riemannian Manifolds and Euclidean spaces, we compare the geometric median shift and mean shift algorithms for clustering synthetic and real data sets.

1 Introduction

Manifold learning attracts more and more attentions in recent years. It can be applied to wide areas, such as manifold clustering [4, 6, 7], which cannot achieve satisfactory results in Euclidean spaces. In particular, there are many works on mean-shift clustering in Euclidean space and Manifolds [8]. The mean shift is also widely applied to computer vision applications, such as feature analysis [1] and image segmentation [5]. Mean shift clustering is a non-parametric clustering algorithm which is based on the nonparametric estimation of a probability density function. The value of the density function at a point can be estimated using the observed samples that fall within a small region around that particular point. A shift window is used for density estimation. Some points can be classified into the same cluster when they converge to the same point in the mean shift iteration process. However, the convergent point may not happen to be one existing element of the dataset. Compared with the mean, the Geometric median is always robust to outliers. Besides, it is the true, existing element in the dataset. This will lead to choosing the different shifting point to be updated between the mean shift and Geometric median shift, as shown in Figure 1.

In general, the median of points often locates on the large density region in the data set, such as geometric median [3] and Tukey median [2]. Similar to mean-shift, the median-shift algorithm in the Euclidean vector space is proposed [6].

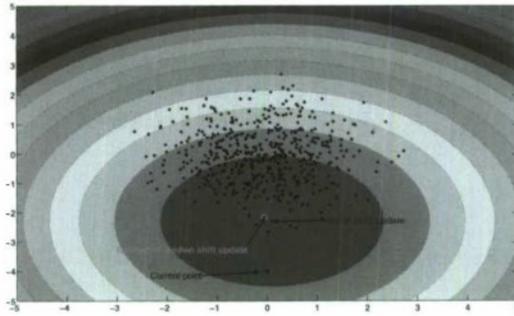


Fig. 1. Comparison between geometric median shift and mean shift. The current point will be respectively updated as the green-black point, by geometric median shift, which is the true point in the original data set, and as the blue point by mean shift, which is not.

Differing from the definition of Geometric median in [3], we define the geometric median as the true point in a data set instead of the non-existing point as presented in [3], which is the least sum of squared distances from the point to others.

Geometric median, however, cannot describe the median of points on a manifold, which frequently occurs in non-vector space [9]. The distances between pair-wise points on a Riemannian manifold cannot be accurately calculated by Euclidean distances, but rather by Geodesic distances. One of the reasons for this lies in different underlying metric spaces between manifolds and Euclidean space. The geodesic distance between two points is equal to either the length of the lines connecting them in Euclidean spaces, or their direct Euclidean distance. This depends on the shape and the metric of the manifolds [11]. Using geometric median, we applied geometric median shift over Riemannian manifolds to clustering in this paper. We make three contributions as follows: i) we propose the new kernel function that calculates Riemannian geodesic distances over Riemannian manifolds. ii) We introduce the geometric median shift vector over Riemannian manifolds. iii) A new geometric median shift algorithm over Riemannian manifolds is also presented. Experiments are reported to demonstrate the performance of our method on synthetic and real data sets. We also make comparisons to other algorithms including mean shift and median shift in Euclidean spaces.

2 Riemannian Metric in Local Coordinates

Define X as a smooth manifold with a Riemannian metric g on X . This means that each point $p \in X$, which defines $g_p(x, y) : T_pX \times T_pX \rightarrow R$ [5], where R is a real number set, T_pX denotes the tangent space of point p in X , and $g_p(x, y)$ is a symmetric, positive definite and bilinear map. In addition, we set v_i and v_j to be the basis of the tangent spaces T_pX at point p as shown in Figure 2.

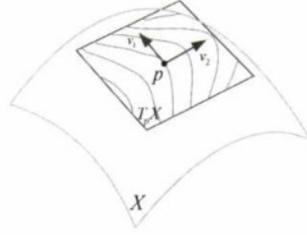


Fig. 2. Tangent space at the point p on a manifold X , with basis vectors v_1 and v_2 in the tangent space

Thus in any local coordinate, a metric is completely determined by the function $g_{ij}(p)$ which may be regarded as the coefficients of a positive definite matrix. Moreover, the length of any piece wise smooth curve $c : [a, b] \rightarrow M$ with $c(a) = p$ and $c(b) = q$ is defined as:

$$\text{length}[c] := \int_a^b \sqrt{g_c(t)(c'(t), c'(t))} dt \quad (1)$$

where $c'(t)$ is the gradient of $c(t)$. On the basis of Eq.1, for any points p and q on manifold. Let $C(p, q)$ denote the space of piecewise smooth curves $c : [a, b] \rightarrow M$ with $c(a) = p$ and $c(b) = q$. We can obtain the distance from any point p and q on a manifold denoted by $d(p, q)$ as

$$d(p, q) = \inf\{\text{length}[c] | c \in C(p, q)\} \quad (2)$$

The distance between a pair of points is defined as the greatest lower bound of the lengths of curves which connect those points. We implement geodesic distance function $d(x, y)$ between points x and y on a manifold using the method in [9] and [10], respectively.

3 Geometric Median-Shift on Riemannian Manifolds

In this section, we introduce geometric median, making a comparison to Tukey median that is the point with the largest tukey depth [2] in the point set. The Euclidean and geodesic distances are applied to the points in Euclidean space and those on manifolds, respectively. Compared with the mean-shift on a Riemannian manifold and Euclidean spaces, the geometric median of points on manifolds is indeed a true point. Besides, points distributed on a Riemannian manifold are always discrete, which cannot be computed by the mean-shift vector on manifolds and Euclidean spaces. In this section, the mean is regarded as a virtual point as opposed to a median point. In other words, when referred to a mean shift iteration, the space is regarded as a continuous one in order to utilize the gradient operator. On the basis of that, we extend each discrete point on a manifold to its neighborhood, and then unite neighborhoods of discrete points contained in the union of open sets.

We start a geometric median shift procedure over Riemannian manifolds from any point distributed on the manifold over the open continuous space. We shift the point along the curves with the special direction. Using the definition of geometric median on a manifold, we propose the kernel density estimate with profile k , bandwidth h and Riemannian metric $d(x, y)$. The geometric median has the property with the minimum sum of squared geodesic distances to other points [3]. We define the kernel density estimate function for geometric median on Riemannian manifolds as

$$\widehat{F}_k(y) = \frac{C_{k,h}}{n} \sum_{i=1}^n k\left(\frac{d^2(y, x_i)}{h^2}\right) \varphi(y) \quad (3)$$

where $k(\cdot)$ is a flat kernel function with value 1, if $0 < x < 1$, and 0 otherwise. $\varphi(y)$ is another kernel function related to the sum of square Riemannian geodesic distances from point y to other points distributed on manifolds. The $C_{k,h}$ and kernel function $\varphi(y)$ are enforced to ensure $\widehat{F}_k(y)$ to be a convex probability density function. The reason why we choose the sum of squared geodesic distances is that it is desirable to do the calculation especially for applying to the large number of points on a manifold as opposed to the geodesic distance between the points on Riemannian manifolds or Euclidean distance in Euclidean spaces.

Theorem 1. $\widehat{F}_k(y)$ is convex if $\varphi(y)$ is a convex function.

Proof. It has been proven in [3] that, a squared geodesic distance to any x_i is convex. With the convex kernel function $k\left(\frac{d^2(y, x_i)}{h^2}\right)$ with the condition that $\varphi(y)$ is a convex function, so $\widehat{F}_k(y)$ can be taken as multiplication between convex kernel functions. It is obvious that $\widehat{F}_k(y)$ is a convex function. ■

Before defining the kernel function $\varphi(y)$ in Eq. 3, we make a comparison between different kernel functions, as shown in Figure 3.

From Figure 3, we choose Gaussian kernel function and define $\varphi(y)$ as.

$$\varphi(y) = \sum_{i=1}^n \exp(-d^2(y, x_i)) \quad (4)$$

where $d(y, x)$ is the geodesic distance between data points y and data point x distributed on Riemannian manifolds. Combining Eq. (3) and Eq. (4), we can get the sum of squared geodesic distances. The final kernel function for geometric median over Riemannian Manifolds is.

$$\widehat{F}_k(y) = \frac{C_{k,h}}{n} \sum_{i=1}^n k\left(\frac{d^2(y, x_i)}{h^2}\right) \exp(-d^2(y, x_i)) \quad (5)$$

The gradient of Eq.5 is.

$$\begin{aligned} \nabla \widehat{F}_k(y) &= \frac{2}{n} \sum_{i=1}^n \left(g\left(\frac{d^2(y, x_i)}{h^2}\right)\right)^{\log_y(x_i)} \exp(-d^2(y, x_i)) \\ &+ \exp(-d^2(y, x_i)) \log_y(x_i) k\left(\frac{d^2(y, x_i)}{h^2}\right) \end{aligned} \quad (6)$$

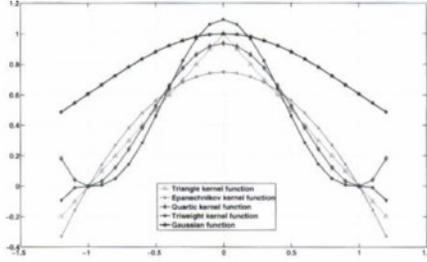


Fig. 3. Comparison of different kernel functions. Triangle kernel function cannot be derivable at the point (0,1). Quartic kernel function is not a strictly monotonous decreasing function. Triweight kernel function does not have a significant change when the x-coordinate of points is close to 1 or -1. Epanechnikov kernel function is negative. Gaussian kernel function has the property with non-negative value, derivable and monotonous decreasing convex.

where $g(x) = -k'(x)$. Further, we denote $\psi(y, x_i)$ as

$$\psi(y, x_i) = \frac{g(\frac{d^2(y, x_i)}{h^2})}{h^2} \exp(-d^2(y, x_i)) + \exp(-d^2(y, x_i))k(\frac{d^2(y, x_i)}{h^2}) \quad (7)$$

On the basis of Eqs. (5) (6) and (7), we give the geometric median shift vector over in tangent spaces and on Riemannian Manifolds in Eq. (8) and Eq. (9), respectively. The computation of the functions $Manifolds_x(y)$ and $\log_x(y)$ are implemented by the method in [9] and [10], respectively.

$$M_{h-tangent}(y) = \frac{\sum_{i=1}^n \psi(y, x_i) \log_y(x_i)}{\sum_{i=1}^n \psi(y, x_i)} \quad (8)$$

$$M_{h-manifold}(y) = Manifolds_y(\alpha M_{h-tangent}(y)) \quad (9)$$

Figure 4 illustrates the meanings of notations used in the above equations.

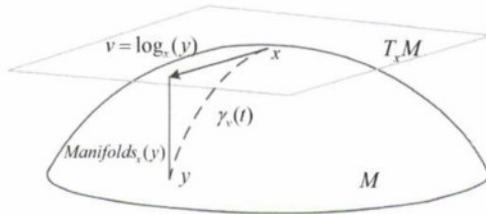


Fig. 4. The function $\log_x(y)$ means the vector that lies in tangent space $T_x M$ at point x . The set $Manifolds_x(y)$ includes the points on the Riemannian manifolds along the curve starting from point x , and the tangent vector of the curve is y at the tangent space $T_x M$.

Similar to the step size used in [3], we set $0 \leq \alpha \leq 2$ to ensure that the point on Riemannian Manifolds can iterate to a converge point. Specifically, we adopt $\alpha = 1$ in our experiments. The final stable point in the original point set that minimizes the sum of squared geodesic distances of other points on a Riemannian manifold is calculated by Eq.(9), is iteratively shifted from some starting points in the point set on the Riemannian manifold.

Theorem 2. The sequence $M_{h\text{-manifold}}(y_k)_{k=1,2,\dots,n}$ generated by successive geometric median shift over Riemannian manifolds converges for all starting locations in point set $\{x_i\}_{i=1,2,\dots,n}$.

Proof. Since $M_{h\text{-manifold}}(y_k)$ and N are finite, the series will converge if there are no cycles, i.e. if $M_{h\text{-manifold}}(y_k) \neq M_{h\text{-manifold}}(y_{k+w})$ for all k and all $w > 0$. According to Theorem 1, $\widehat{F}_k(y)$ is a convex function. This is because $\exp(-d^2(y, x_i))$ is a convex function. We then have

$$\widehat{F}_k(y_{k+1}) - \widehat{F}_k(y_k) \geq \nabla \widehat{F}_k(y_k)(y_{k+1} - y_k). \tag{10}$$

the geometric median y is

$$y = x_{i^*}, \text{ where } i^* = \arg \min_{y \in \{x_i\}_{i=1,\dots,n}} \sum_{i=1}^n d^2(y, x_i) \tag{11}$$

Therefore we have

$$\widehat{F}_k(y_{k+1}) - \widehat{F}_k(y) \geq \sum_{i=1}^n \nabla \widehat{F}_k(d^2(y, x_i))(d^2(y_{k+1}, x_i) - d^2(y_k, x_i)) \tag{12}$$

If $y_{k+1} = \text{Manifolds}_y(\alpha M_{h\text{-tangent}}(y_k))$, then we have

$$\sum_{i=1}^n d^2(y_{k+1}, x_i) > \sum_{i=1}^n d^2(y_k, x_i). \tag{13}$$

Eq. (13) can be re-written as

$$\sum_{i=1}^n \nabla \widehat{F}_k(d^2(y, x_i))(d^2(y_{k+1}, x_i) - d^2(y_k, x_i)) > 0. \tag{14}$$

From the inequalities Eq. (12) and Eq. (14), we can deduce that the inequality of $\widehat{F}_k(y_{k+1}) > \widehat{F}_k(y_k)$ is true for the sequence $\{y_0, y_1, \dots, y_k\}$ generated from Eq. (9). The value in the corresponding sequence $\{\widehat{F}_k(y_0), \widehat{F}_k(y_1), \dots, \widehat{F}_k(y_k)\}$ is strictly increasing. This leads to $\widehat{F}_k(y_{k+w}) > \widehat{F}_k(y_k)$ for all $w > 0$, and therefore we have $y_{k+w} \neq y_k$. ■

Based on the above theorems, the Geometric median shift algorithm over a Riemannian manifold is given in **Algorithm 1**.

Algorithm 1. Geometric median shift over Riemannian manifolds

Require: Input: Points on Riemannian manifolds x_i ($i = 1, \dots, n$), ε ,
bandwidth h and α

Output: Distinct local modes

Extend all the points to the union of all their neighborhoods to form a smooth and continuous point space, set $k=1$, and check if $0 \leq \alpha \leq 2$

Ensure: for $i \leftarrow 1, \dots, n$

$y \leftarrow x_i$

do

$$M_{h-tangent}(y) = \frac{\sum_{x_i \text{ within a window}} \psi(y, x_i) \log_y(x_i)}{\sum_{x_i \text{ within a window}} \psi(y, x_i)} \quad (\text{Eq. (8)})$$

$$y \leftarrow \text{Manifolds}_y(\alpha M_{h-tangent}(y)) \quad (\text{Eq. (9)})$$

until $\|M_{h-tangent}(y)\| \leq \varepsilon$

$$y_k = x_{i^*}, \text{ where } i^* = \arg \min_{y \in \{x_i \text{ within a window}\}} \sum d^2(y, x_i) \quad (\text{Eq. (11)})$$

Retain y_k as a local mode

$k \leftarrow k + 1$

end for

Theorem 3. The time complexity of our algorithm for finding the geometric median of data points is $O(n^2)$, where n is the number data points in a set.

Proof. Let us examine the algorithm that includes two main steps.

Step 1: Calculating the sum of distances to other $n-1$ points for each point needs $O(n-1)$. Therefore, it takes $O(n * (n-1))$ to get the sum distances of all points in a data set.

Step 2: It is obvious that selecting the points that has the minimum sum of distances to others as the geometric median of data points set needs $O(n)$. The time complexity needs $O(n) + O(n * (n-1)) = O(n^2)$. ■

4 Experiments

We have implemented the Geometric Median-shift algorithm on manifolds in C++ and Matlab. A number of experiments are performed to evaluate the performance of our algorithm. Note that the orientation of curves on a manifold will not be considered in the experiments.

4.1 Geometric Median Shift on Synthetic Datasets

In Figure 5, we compared our method to the mean-shift over Euclidean space, over Riemannian manifolds, and over both Euclidean space and Riemannian manifolds, respectively.

Without surprise, the resulting clusters vary in size, shape and so on, as shown in Figure 6. Only is our proposed algorithm able to find the correct clusters.

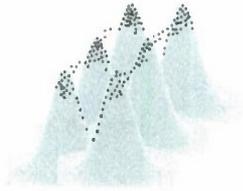


Fig. 5. 800 points distributed on the neighborhood of 6 peaks of a surface



Fig. 6. The comparison of Geometric Median-shift and Mean-shift clustering on points over manifold and Euclidean space. (a) 6 resulting clusters have been found. This is because the geodesic distance and a geometric median as a true point are employed. (b) and (d) show the comparison between the processes of applying the geometric median shift to points in Euclidean space. The median and mean of points represent the true point in the original data set and non-existing point. The points on manifold and Euclidean space are measured by the geodesic distance and Euclidean distance, respectively. These two facts lead to different cluster results on the Geometric median shift and mean shift over Riemannian manifold and Euclidean space. (c) 3 clusters were formed. This is because the mean of points can be a virtue point corresponding to the process of geometric median shift.

4.2 Geometric Median Shift on Real Datasets

We applied our algorithm to cluster the data points of 4 swiss-roll type data sets [6], each swiss roll with about 500 data points distributed on a manifold as shown in Figure 7. The clustering results using mean shift algorithm via Euclidean distance are not as good as those using geometric median shift over Riemannian manifold via Riemannian geodesic distance in [10].

Furthermore, we compare geometric median shift algorithm with the mean shift using different h values in Eq.(3) by testing 4 swiss rolls data sets. The chosen validation metric for evaluating our clustering results is the average Euclidean distance to each cluster center. The smaller the average Euclidean distance is, the better the clustering results are. Without surprise, the average distance between the center and the members of each cluster formed by our method is smaller than that formed by mean shift. This is because the geodesic distance characterizes the data distribution better than the Euclidean distance for the swiss roll data sets. The two results are reported in Tables 1 and 2, respectively. Note that outliers are excluded for computing the distances to cluster centers in following experiments.

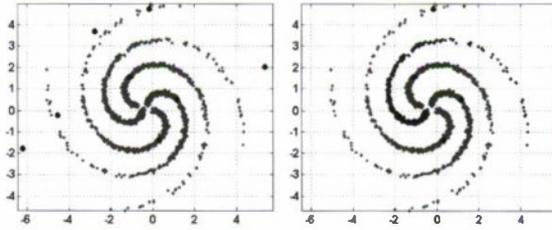


Fig. 7. Resulting clusters using Mean shift clustering result (left) in Euclidean space and geometric median shift (right) over Riemannian manifolds using Riemannian geodesic distances in [10] with window size 1×1

Table 1. Clustering results of geometric median shift using Riemannian geodesic distance in [10] tested on 4 swiss rolls in [6]

h	window size	#cluster	Average distance to each cluster center		
			minimum	maximum	average
2	2×2	4	3.6582	3.8344	3.7214
3	3×3	4	3.6576	3.9982	3.7152
4	4×4	4	3.6827	3.8991	3.7921
5	5×5	4	3.6119	4.0017	3.9133
6	6×6	4	3.6772	3.9982	3.7287
7	7×7	4	3.6225	3.8256	3.7821
8	8×8	4	3.6776	3.7815	3.6988
9	9×9	4	3.6852	3.7881	3.6879
10	10×10	4	3.6684	3.7751	3.6693

Table 2. Clustering results of mean shift using Euclidean distance tested on 4 swiss rolls in [6]

h	window size	#cluster	Average distance to each cluster center		
			minimum	maximum	average
2	2×2	2	4.0582	4.5844	4.3213
3	3×3	1	4.1265	4.1265	4.1265
4	4×4	1	4.1394	4.1394	4.1394
5	5×5	1	4.1442	4.1442	4.1442
6	6×6	1	4.1492	4.1492	4.1492
7	7×7	1	4.1699	4.1699	4.1699
8	8×8	1	4.1742	4.1742	4.1742
9	9×9	1	4.1764	4.1764	4.1764
10	10×10	1	4.1791	4.1791	4.1791

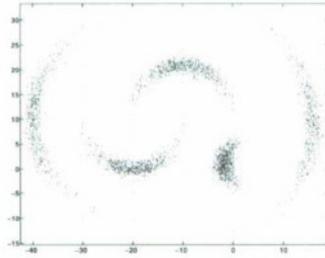


Fig. 8. The results of visualizing the five crescents data sets in [6]

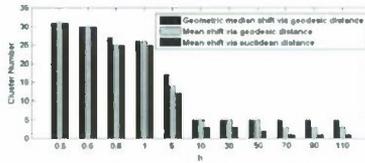


Fig. 9. Cluster number of three different algorithms tested on 5 crescents data sets

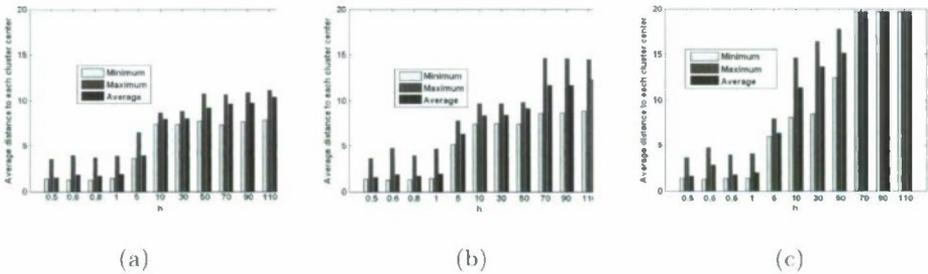


Fig. 10. Average sum-of-distance for each cluster of Geometric median shift via Riemannian geodesic distance (a), mean shift via Riemannian geodesic distance (b) and mean shift via Euclidean distance (c)

Using geometric median shift and mean shift algorithms over Riemannian manifolds via Riemannian manifold distance in [10], as well as mean shift in Euclidean distances. We tested five crescents data sets [6] (see Figure 8), which contain 5053 data points with different sizes of parameter h in Eq.(3). The clustering results in terms of both the average distance of the center to each cluster center and the running time are listed. We also calculated the average sum-of-distance for each cluster and running time with with different sizes of parameter h in Eq.(3), as shown in Figures 10 and 11, respectively.

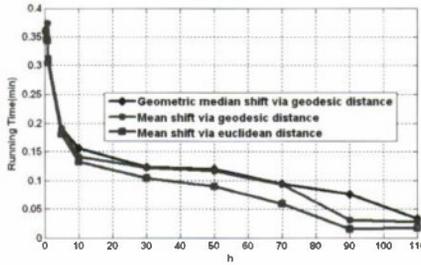


Fig. 11. Running time of three different algorithms tested on 5 crescents data sets

4.3 Results Analysis

As shown in Figure 1, geometric median shift is always iterated to next existing point rather than a non-existing point in mean shift. The bigger value of Eq.(5), the higher likely it will converge to the current existing point. This happens in a case where there are the higher number of points around the current point, and the smaller sum of squared distance from the current point to all others within a window. This is different from the mean shift, in which a mean is defined as the gravity of points. The mean may not thereby be a true point, so the algorithm will continue to converge. Due to this reason, the mean shift always produces the smaller number of clusters than Geometric median shift does, such as the examples shown in Figure 9. Mean shift commonly groups the points even with the relatively large geodesic distances through some non-existing points. So its average distance to cluster centers is larger than one by geometric median shift. This fact has been validated by results reported in Figure 10. For all these reasons, in terms of average sum-of-distances, geometric median shift is able to produce cluster results that are better than the mean shift algorithm, particular for date sets with implicit manifolds. It takes $O(n)$ time to obtain the mean of points while the calculation of the Geometric median takes $O(n^2)$. Because of this, the running time of the mean shift algorithm is less than that of geometric median shift both on manifolds and Euclidean distances. In general, calculating the geodesic distance takes more time than Euclidean distance between two data points. So the geometric median shift over Riemannian manifolds spends more time than others, which leads to the results in Figure 11. Further, if the shifting window size is increased, the number of iteration will be decreased. The running time is also reduced accordingly. In addition, the increase of the size of the shift window will make the more number of outliers to become the members of clusters. This makes the average sum-of-distance bigger, as shown in Figure 10. This fact is true for both Geometric median shift and mean shift algorithms.

5 Conclusion

Manifold clustering attracts more and more attentions in recent years. In this paper, we have presented a geometric median shift algorithm for clustering

data points on Riemannian manifolds. Given two data points, their Riemannian geodesic may not equal to their corresponding Euclidean distance. This fact may lead to forming different clustering results by using the mean shift. From the experiments, we conclude that the clustering results by using the true median point on a manifold are more accurate than those by the mean shift in Euclidean space. Furthermore, compared to using Tukey median, our algorithm for calculating the geometric median reduces the complexity from $O(n^2 \log n^2)$ to $O(n^2)$. Applying the proposed approach to more applications is our future work.

References

1. Comaniciu, D., Meer, P.: A robust approach toward feature space analysis. *IEEE Transaction on Pattern Analysis and Machine Intelligence* 24(5), 603–619 (2002)
2. Cuesta-Albertos, J.A., Nieto-Reyes, A.: The random tukey depth. *Computational Statistics and Data Analysis* 52(11), 4979–4988 (2008)
3. Fletcher, T., Venkatasubramanian, S., Joshi, S.: Robust statistics on manifolds via the geometric median. In: *CVPR*, pp. 1–8 (2008)
4. Gu, Q., Zhou, J.: Co-clustering on manifolds. In: *ACM SIGKDD*, pp. 359–368 (2009)
5. Pooransingh, A., Radix, C., Kokaram, A.: The path assigned mean shift algorithm: a new fast mean shift implementation for colour image segmentation. In: *ICIP*, pp. 597–600 (2008)
6. Shapira, L., Avidan, S., Shamir, A.: Mode-detection via median shift. In: *ICCV*, pp. 1–8 (2009)
7. Souvenir, R., Pless, R.: Manifold clustering. In: *ICCV*, pp. 648–653 (2005)
8. Subbarao, R., Meer, P.: Nonlinear mean shift for clustering over analytical manifolds. In: *CVPR*, pp. 1168–1175 (2006)
9. Subbarao, R., Meer, P.: Non-linear mean-shift over riemannian manifold. *IJCV* 84(1), 1–20 (2009)
10. Tenenbaum, J., de Sliva, V., Langford, J.: A Global Geometric Framework for nonlinear Dimensionality Reduction. *Science* (2000)
11. Tuzel, O., Porikli, F., Meer, P.: Human detection via classification on riemannian manifolds. In: *CVPR*, pp. 1–8 (2007)

Multi-manifold Clustering

Yong Wang^{1,2}, Yuan Jiang², Yi Wu¹, and Zhi-Hua Zhou²

¹ Department of Mathematics and Systems Science
National University of Defense Technology, Changsha 410073, China
yongwang82@gmail.com, wuyi_work@sina.com

² National Key Laboratory for Novel Software Technology
Nanjing University, Nanjing 210093, China
{jiangy, zhoush}@lamda.nju.edu.cn

Abstract. Manifold clustering, which regards clusters as groups of points around compact manifolds, has been realized as a promising generalization of traditional clustering. A number of linear or nonlinear manifold clustering approaches have been developed recently. Although they have attained better performances than traditional clustering methods in many scenarios, most of these approaches suffer from two weaknesses. First, when the data are drawn from hybrid modeling, i.e., some data manifolds are separated but some are intersected, existing approaches could not work well although hybrid modeling often appears in real data. Second, many approaches require to know the number of clusters and the intrinsic dimensions of the manifolds in advance, while it is hard for the user to provide such information in practice. In this paper, we propose a new manifold clustering approach, *mumCluster*, to address these issues. Experimental results show that the performance of the proposed *mumCluster* approach is encouraging.

1 Introduction

Traditional clustering methods, such as K -means [1], are based on the idea that a cluster is centered around a single point when measuring similarity. Recently, a large number of research efforts have shown that the perceptually meaningful structure of the points possibly resides on a low-dimensional manifold [2,3]. Therefore, regarding cluster as a group of points around a compact manifold becomes a reasonable and promising generalization of traditional clustering, leading to *manifold clustering* [4].

Roughly speaking, the research on manifold clustering can be classified into two branches, i.e., linear and nonlinear. Generalized Principal Component Analysis (GPCA) [5,6] and K -planes [7,8,9] assume the samples to be well approximated by a mixture of affine subspaces (or linear manifolds). However, manifolds in natural data are generally nonlinear in the original space [2]. Spectral clustering (SC) [10,11] is a good option when the samples are lying on separated clusters where each cluster contains points sampled from a single nonlinear manifold. Alternatively, Cao and Haralick [12] use the local dimension and mean square error to infer clusters. However, when there are intersections among clusters, their performance will degenerate. K -manifolds [4] is primarily motivated to cluster samples generated from intersecting nonlinear manifolds, which will fail when the clusters are widely separated.

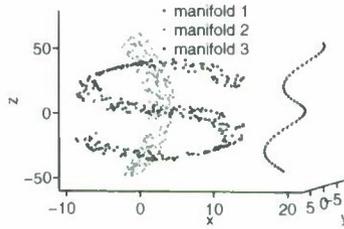


Fig. 1. Data points drawn from a hybrid modeling

There are two main difficulties for existing methods. On the one hand, they usually work well either in separated case or in intersecting case. When the input data points are drawn from a hybrid modeling (see Figure 1) where some manifolds are separated, while some others are intersected with each other, the quality of clustering degenerate. On the other hand, many of existing methods require the user to provide the number of clusters and their intrinsic dimensions in advance, while such information are difficult to be given in practice. For example, considering a data set consisting of face images of different individuals under various lighting conditions, it is difficult for the user to know whether the underlying manifolds are separated or intersected, as well as the number of clusters and the intrinsic dimensions ahead. Thus, to enable manifold clustering to deal with more real tasks, it is important to design manifold clustering approaches which are able to work well when the samples are drawn from hybrid modeling, and which can adaptively determine the number of clusters and dimensions.

In this paper, we propose a new manifold clustering method called *mumCluster* (MUlti-Manifold Clustering). Our basic idea is based on the observation that if we can make the constructed undirected graph in spectral clustering more faithful, i.e., data points belonging to different manifolds will not be connected, then spectral clustering can be used to identify different manifolds accurately. Thus, our scheme first identifies the separate subsets of the original data, and then determines whether a subset is composed of a single manifold or intersecting manifolds. For each intersecting subset, we will exclude the influence of the inaccurate connected relationships among different manifolds. Finally, spectral clustering is used to further infer clusters. Moreover, a strategy is developed to automatically determine the number of manifold clusters and their corresponding dimensions.

The rest of this paper is organized as follows: Section 2 briefly reviews the related manifold clustering methods. In Section 3, the *mumCluster* method is presented, followed by a strategy to determine the number of clusters and their dimensions. Computational complexity analysis of the proposed method is also presented in this section. In Section 4, we experimentally evaluate the performance of our proposed method using synthetic and real-world data. Section 5 concludes this paper.

2 Related Work

Cluster analysis [13] seeks to group internally similar objects into the same cluster while dissimilar objects into different clusters. Traditional clustering methods, such as

K -means [1], assume the data are centered around some prototypes. They could not separate clusters that are nonlinearly separable or centered around manifolds.

GPCA [5,6] and K -planes [7,8,9] are representative linear manifold clustering methods. GPCA models the underlying manifolds with a set of homogeneous polynomials, then the constructed models are used to infer clusters. Alternatively, K -planes addresses linear manifold clustering by iterating between assigning data to manifolds, and modeling a manifold to each cluster. Although successful for mixtures of linear clusters, both of them fail to deliver good performance in the presence of nonlinear structures (e.g., Figure 3 (a) and (b)). Since nonlinear methods can also work well on linear clusters, in this paper, we focus on the nonlinear manifold clustering.

Spectral clustering [10,11] is a good option for nonlinear manifold clustering when samples are generated from separated clusters where each cluster contains data points from a single manifold [14]. However, when there are intersections in some areas, spectral clustering could not work well (e.g., Figure 3 (c)). The reason is that the performance of spectral clustering is heavily relied on the constructed undirected graph, different clusters near a manifold intersection will easily be connected by the undirected graph, thus diffusing information across the wrong manifolds [15]. K -manifolds [4] groups data lying on intersecting nonlinear manifolds, which begins by estimating geodesic distances between points, then an expectation maximization (EM) type strategy is used to iterate between estimating the manifolds using node-weighted MDS and assigning each point to the specified manifolds. Unfortunately, the estimation of geodesic distances fails when there are separated clusters, leading to incorrect clustering (e.g., Figure 3 (d)). The method most related to ours was proposed by Cao and Haralick [12], which groups neighboring points into a cluster if they have the same local dimension and the mean square error of representing the new cluster is small. This method can handle the hybrid modeling to some extent, by using graph methods to identify different connected components. However, it is primarily based on the local dimension, thus the method usually treats the intersections as clusters since the local dimension in the intersections are higher than the other areas (e.g., Figure 3 (e)).

3 MumCluster

Given a set of data points $X = \{x_i \in \mathbb{R}^D, i = 1, 2, \dots, N\}$ sampled from $k > 1$ distinct manifolds $\{\Omega_j \subseteq \mathbb{R}^D, j = 1, 2, \dots, k\}$ with dimension $d_j = \dim(\Omega_j)$, $0 < d_j < D$. The samples are unorganized, i.e., we do not know which points belong to which manifold. Moreover, some manifolds are intersected with each other which form intersecting manifolds. Our objectives are:

1. Identify the number of manifolds k and their intrinsic dimensions $\{d_j, j = 1, 2, \dots, k\}$;
2. Partition the given samples into the manifold(s) they belong to.

Though a considerable amount of work has been done in this field, as we have reviewed before, they could not work well on the hybrid modeling. Moreover, many of them need the user to specify k and $\{d_j, j = 1, 2, \dots, k\}$. In what following, we propose the mumCluster method to address these issues.

Our main strategy is trying to construct more faithful undirected graph in spectral clustering, i.e., data points belonging to different manifolds will not be connected.

Therefore, *mumCluster* designs a “divide and conquer” strategy to realize this purpose. This scheme first divides the complicated intersecting manifolds from the single manifolds, then each intersecting subset is further divided into intersection areas and non-intersection areas. More attention is paid to the intersection areas, where many of the inaccurate connected relationships situated. The details of the method are presented in Subsection 3.1, followed by a strategy to automatically determine the number of clusters and their dimensions in Subsection 3.2. Complexity analysis is presented in Subsection 3.3.

3.1 To Deal with Hybrid Modeling

Generally, hybrid modeling can be divided into different connected subsets, with some subsets containing only single manifold, while the others containing intersecting manifolds. To deal with the two different structures separately, we propose to use spectral clustering to partition the samples coarsely into different connected subsets. Generally, there are different versions of spectral clustering. Following von Luxburg’s suggestion [14], the following unsymmetrical normalized spectral clustering [10] is adopted:

1. Constructing a similarity graph G : Put an edge between node i and j if i is among L nearest neighbors of j , and vice versa.

2. Determining the weighted matrix W : If node i and j are connected, then put a weight w_{ij} as $w_{ij} = 1$ (simple weight); otherwise, put $w_{ij} = 0$.

3. Spectral decomposition: Compute the first r eigenvectors u_1, u_2, \dots, u_r , corresponding to the r smallest eigenvalues, of the generalized eigenproblem $Eu = \lambda Fu$, where F is a diagonal matrix with $F_{ii} = \sum_j w_{ij}$ and $E = F - W$. Let $U = [u_1, u_2, \dots, u_r] \in \mathbb{R}^{N \times r}$.

4. Clustering by K -means: Group the points $y_i, i = 1, 2, \dots, N$ into r clusters using K -means, where y_i is the vector corresponding to the i -th row of U .

In the above procedure, r should be provided. We will discuss on how to decide r in the next subsection.

After the different connected subsets $\Xi_c, c = 1, \dots, r$ have been identified, the problem is how to determine their structure, i.e., single or intersecting. For this purpose, our basic idea is to resort to the intrinsic dimension id . It is based on the observation that if samples come from a single manifold, then the intrinsic dimension of each point on this manifold should be the same; otherwise, they are different. Details on estimating id will be presented in the next subsection.

If the connected subset consists of a single manifold, then a manifold cluster has been revealed. However, for each intersecting subset Ξ^{is} , further procedures are needed to reveal different manifold clusters. The first should be to identify the intersection areas Π^{ia} and the non-intersection areas Π^{nia} . Generally, the points in Π^{ia} have higher dimension than the other parts. Therefore, the points with the highest dimension d_{\max} should be first grouped into Π^{ia} . In practice, the structure in the intersection area is usually complex. To ensure this area to be identified accurately, the ε -neighbors can be used. That is,

$$x \in \Pi^{ia}, \quad \text{if} \quad \|x - x^{ip}\|^2 < \varepsilon, \quad (1)$$

where x^{ip} is any point with dimension d_{\max} . Finally, Ξ^{is} is divided into Π^{ia} and Π^{nia} .

The points in Π^{ia} and Π^{nia} may consist of many small clusters (called *intersection clusters* and *non-intersection clusters*, respectively), which should be grouped in order to tackle them separately. Generally, these clusters are unconnected, thus spectral clustering can still be used here to group them. If the dimensions on some non-intersection clusters are different, it implies that there may still exist some other intersection clusters with lower d_{\max} . Therefore, we should go back to identify these areas until there is no hidden intersection.

The intersection area implies that there are different manifolds passing across each other which should be revealed. Though, the manifold clusters are nonlinear, each intersection cluster can be considered as a mixture of manifolds with linear structure since it is a local area. Thus, K -planes can be adopted to reveal the different manifolds (named *fine clusters*) in each intersection cluster. Specifically, given the number of clusters k^* and the dimensions $d_1^*, d_2^*, \dots, d_{k^*}^*$.

1. Initialization: Assign each point to a cluster randomly to give an initial partition $\{C_1^*, C_2^*, \dots, C_{k^*}^*\}$. Then, alternating between the following two steps until convergence.
2. Cluster update: Find a center μ_i^* and a set of bases $\Phi_i = [\varphi_1^i, \varphi_2^i, \dots, \varphi_{d_i^*}^i]$ for cluster C_i^* such that the reconstruction error is minimum.
3. Cluster assignment: For each point x_m^* in the considered intersection cluster, determine the space j such that

$$\begin{aligned} & (x_m^* - \mu_j^*)^T (I - \Phi_j \Phi_j^T) (x_m^* - \mu_j^*) \\ & = \min_{i=1, \dots, k^*} (x_m^* - \mu_i^*)^T (I - \Phi_i \Phi_i^T) (x_m^* - \mu_i^*), \end{aligned} \tag{2}$$

where I is an identity matrix. Then, x_m^* is assigned to the j -th cluster C_j^* .

As indicated before, the constructed undirected graph for each intersecting subset may connect different manifolds, making the partition of samples into the manifold they belong to impractical. To reveal different manifolds, the connections between them should be cut out, and should be preserved among the same manifold. Since the unfaithful connections mainly come from the different fine clusters, we cut the connections among them, while connect all the points in the same fine cluster to preserve the manifold structure. Finally, a new undirected graph G_{new} is obtained for each intersecting subset Ξ^{is} . Thus, spectral clustering is used to finally group points in each Ξ^{is} into different manifold clusters.

3.2 To Determine the Number of Clusters and the Intrinsic Dimensions

Hereinbefore, we have shown our scheme to partition the given samples into the manifold they belong to. However, it is based on the given number of clusters and their intrinsic dimensions, and how to adaptively determine these parameters are not resolved. In the following, we propose to use cigengap, local intrinsic dimension estimator and a new bottom-up search procedure to address these issues.

First, as demonstrated in [14], the number of connected components r in the adopted spectral clustering equals the multiplicity r of the eigenvalue zero of the generalized eigen-problem. Therefore, r can be determined by using the *eigengap* heuristic. That is,

$$\text{if } |\lambda_l - \lambda_{l-1}| \leq 10^{-6} < |\lambda_{l+1} - \lambda_l|, \quad \text{then } r = l, \quad (3)$$

where 10^{-6} is used to replace zero to avoid numeric problem.

The intrinsic dimension id of each point can be estimated by using a local dimension estimator. It is based on the observation that though the manifold structures are globally nonlinear, they are locally linear [3]. Moreover, it is known that the first id largest eigenvalues of the covariance matrix are significantly higher than the others and thus can be used as an estimation to the intrinsic dimension, when the original data are sampled from an id -dimensional manifold [16]. In more detail, we can estimate the intrinsic dimension by:

1. Calculate the local covariance matrix: For each point x_i , find its L nearest neighbors x_i^1, \dots, x_i^L , then calculate the local covariance matrix

$$C_i = 1/L \sum_{j=1}^L (x_i^j - \mu_i)(x_i^j - \mu_i)^T, \quad (4)$$

where $\mu_i = 1/L \sum_{j=1}^L x_i^j$ is the mean vector.

2. Intrinsic dimension estimation: Determine the sorted eigenvalues $\lambda_1^i \geq \dots \geq \lambda_D^i$ of C_i ,

$$\text{if } \lambda_j^i / \lambda_1^i < 0.05 \leq \lambda_{j-1}^i / \lambda_1^i, \quad \text{then } id = j - 1. \quad (5)$$

More challenging is to determine k^* and $d_1^*, d_2^*, \dots, d_{k^*}^*$ in the K -planes algorithm which is used to reveal fine clusters in each intersection cluster. Our solution is based on a bottom-up search strategy, which starts from the lowest dimension d_{\min} . Moreover, we can determine the possible dimensions and the number of clusters, which reduce the search space. First, let us introduce the following notion.

Definition: Effective Dimension (ED) [17]

Given k subspaces $\Omega = \bigcup_{i=1}^k \Omega_i$ in \mathbb{R}^D of dimension $d_i < D$, and N_i sample points $X_i = \{x_i^j, j = 1, \dots, N_i\}$ drawn from each subspace Ω_i , the effective dimension is defined to be:

$$ED(X, \Omega) \triangleq 1/N \sum_{i=1}^k d_i(D - d_i) + 1/N \sum_{i=1}^k N_i d_i. \quad (6)$$

Effective dimension $ED(X, \Omega)$ is the “average” numbers needed to assign to per sample of X . Generally, there could be many manifold structures Ω which can fit X , while the manifold structure that leads to the minimum ED normally corresponds to an “efficient” and hence “natural” interpretation of the data, see [17]. Formally, ED is low if the number of clusters and dimension of each cluster are small. Therefore, to faithfully fit the underlying manifold structure, we should search for the structure which minimizes ED among all possible structures under certain criterion. To be consist with the K -planes algorithm, the reconstruction error is a good choice.

```

mumCluster( $X, L, \varepsilon, \zeta_{\max}$ )


---


Input:
 $X$ :  $D \times N$  feature matrix
 $L$ : number of nearest neighbors
 $\varepsilon$ : threshold for determining the intersection area
 $\zeta_{\max}$ : maximum error threshold
Process:
1 Construct graph  $G$  with weighted matrix  $W$ 
2 Group using spectral clustering on  $W$  with eigengap
3 for each connected subset
4   Compute the intrinsic dimension  $id$  for each point
5   if  $id$ 's are the same
6     Output this connected subset as a cluster
7   else
8     Construct a new graph  $G_{new}$ 
9     Group using spectral clustering on  $G_{new}$ 
10  endif
11 end
Output:
 $\{C_1, C_2, \dots, C_k\}$ : the results of clustering

```

Fig. 2. Pseudo-code of the mumCluster method

To reduce the search space, the following observation is considered: the intersection clusters are crossed by different manifolds, moving continuously from the non-intersection clusters. Suppose an intersection cluster is connected with m non-intersection clusters, then the dimensions of the non-intersection clusters imply the possible dimensions of the fine clusters, while the number of non-intersection clusters limits the number of fine clusters.

Our bottom-up strategy can be summarized as follows:

1. For each intersection cluster, determine the number of connected non-intersection clusters (i.e., m) and the dimension of each non-intersection cluster (i.e., d_1, \dots, d_m);
2. Suppose there are n different sorted numbers in $\{d_1, \dots, d_m\}$, i.e., $d^1 < \dots < d^n$. Assign the possible number of clusters to the range from n to m . For each specified number, the dimension for each cluster is given by one number in $\{d^1, \dots, d^n\}$ starting from the lowest to the highest, and at least one cluster has dimension d^j , $j = 1, \dots, n$.
3. For each given number and dimensions of the clusters, compute its ED if the reconstruction error by K -planes is smaller than a specified maximum error ζ_{\max} . Otherwise, ED is set to be the maximum number $N_{\max} = 100$.
4. The best number of clusters and their dimensions are given by the structure with the minimum ED.

Our proposed mumCluster reveals that there are three intersection clusters for the points sampled from Figure 1, where each cluster is connected with $m = 4$ non-intersection

Table 1. Effective dimension (ED) for each intersection cluster in Figure 1 w.r.t the possible structure (the best is marked in boldface)

STRUCTURE	2	(2,2)	(2,2,2)	(2,2,2,2)
INTERSECTION CLUSTER 1	100	2.021	2.031	2.041
INTERSECTION CLUSTER 2	100	2.019	2.029	2.039
INTERSECTION CLUSTER 3	100	2.020	2.030	2.040

clusters. The possible structure (in the form of $(d_1^*, d_2^*, \dots, d_k^*)$ for k^* clusters) and their corresponding effective dimension are tabulated in Table 1.

Figure 2 shows the Pseudo-code of mumCluster.

3.3 Complexity Analysis

The computational complexity of our proposed mumCluster is dominated by three parts: intrinsic dimension estimation, connected components search and fine clusters identification. Intrinsic dimensions of N D -dimensional data points are estimated by performing local PCA on L nearest neighbors of each point, the complexity is $N \times O(LD \min(L, D))$. Spectral clustering is used to search for the r connected components, with the total complexity $O((D + L + r)N^2 + Nr^2t)$, where $O((D + L)N^2)$ stands for the time complexity of constructing similarity graph, $O(rN^2)$ stands for the complexity of computing the first r generalized eigenvectors and $O(Nr^2t)$ is the complexity of K -means in r -dimensional space for t iterations. Since $r \ll N$, $L \ll N$ and K -means converges very quickly, the complexity of connected components search is limited by $O(N^2 \max(D, N))$. The complexity analysis of grouping fine clusters using K -planes is not straightforward, since we do not know the exact number of points to be grouped and a bottom-up scheme as shown in Subsection 3.2 is needed to automatically determine the number of clusters and their dimensions. However, following the same analysis in [8], the overall worst case time complexity (an upper bound) of this procedure is $O(m^2) \cdot O(DN \min(D, N))$ when there are m non-intersection clusters. Note that, this result does not reflect its real running time as demonstrated by the experiments presented in the next section. To sum up, the computational complexity of mumCluster is limited by $O(N^2 \max(D, N))$ in total, which is determined by the number of data points and the number of features.

4 Experiments

We now evaluate the performance of our mumCluster using synthetic data and real data. Note that the number of manifold clusters and their dimensions are provided for all the other manifold clustering methods except for mumCluster. For spectral clustering (SC), the unsymmetrical normalized spectral clustering [10] is used.

4.1 Hybrid Modeling Data

The hybrid modeling data shown in Figure 1 are drawn from one helix, one swiss-roll, and one two-dimensional surface in \mathbb{R}^3 . The number of points are 200, 1000, 600,

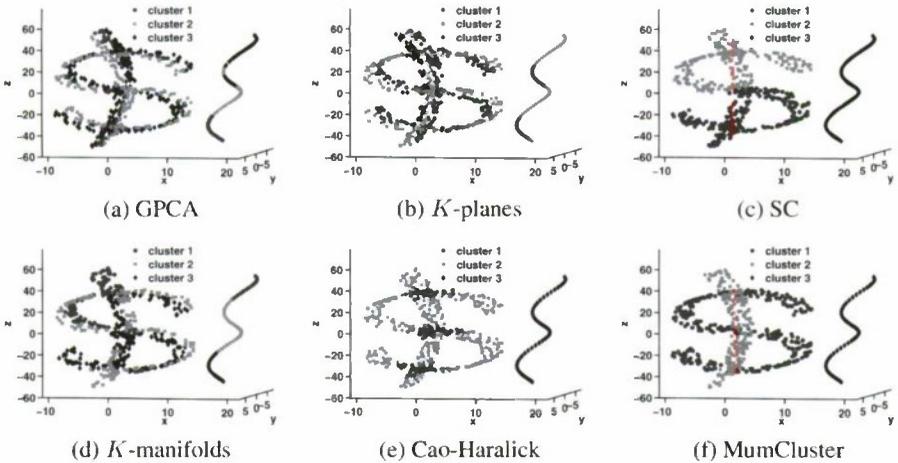


Fig. 3. Grouping results using different manifold clustering methods

Table 2. Clustering accuracy (%) of the different methods on the hybrid modeling data

GPCA	K -PLANES	SC	K -MANIFOLDS	CAO-HARALICK	MUMCLUSTER
38.11	40.06	57.39	40.39	60.17	99.06

respectively. As we can see from Figure 3, all the other methods do not work well on this data set. Table 2 reports the clustering accuracy of the different methods. Obviously, our method performs quite well. GPCA and K -planes do not work well in this nonlinear case because of their linear nature, while the method of Cao and Haralick treats the intersections as clusters. SC diffuses wrong clustering information across the intersecting manifolds, while K -manifolds fails to estimate faithful geodesic distances when there are separated clusters.

4.2 Single Modeling Data

It is interesting to compare our mumCluster with SC on data containing multiple single manifolds, and compare with K -manifolds on data containing intersecting manifolds, where SC and K -manifolds can work well, respectively. It is easy to see that when points are sampled from multiple separated single manifolds, our mumCluster is in fact as same as SC and therefore the results are not presented here due to the space limit. In the following, we compare mumCluster with K -manifolds on data containing intersecting manifolds. The spirals data set¹ (see Figure 1 of [4]) where K -manifolds can work well is used for the comparison. We run mumCluster and K -manifolds over five random samplings from this evaluated data set, as well as the other methods which can be used for intersecting manifolds. Table 3 reports the clustering accuracy. The results demonstrate that mumCluster generally outperforms the other methods.

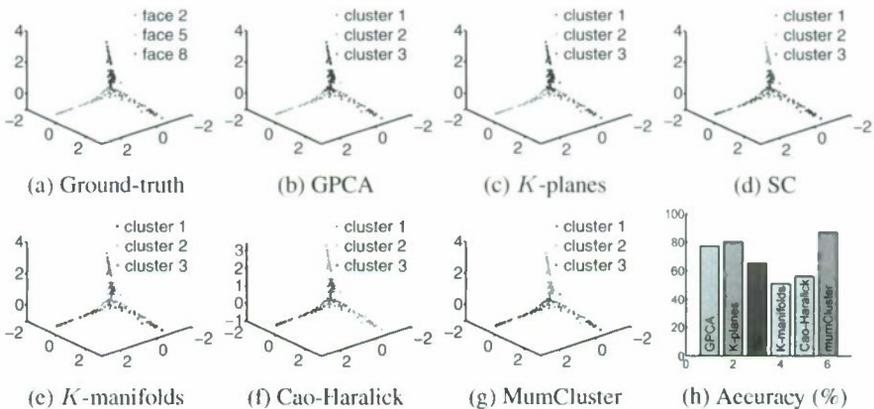
¹ <http://www.cs.wustl.edu/~rms2/kmanifolds.htm>

Table 3. Clustering accuracy (%) over five random samplings from the spirals data set

DATA SET	A	B	C	D	E
GPCA	48.8	42.4	43.6	44.8	47.0
K -PLANES	48.2	40.6	49.4	46.4	46.4
CAO-HARALICK	52.0	50.6	47.6	51.0	48.4
K -MANIFOLDS	98.0	96.0	97.6	97.6	96.6
MUMCLUSTER	100.0	99.8	100.0	99.6	99.2

4.3 Illumination Variant Face Clustering

In this experiment, the face images in the Yale Face Database B² [18] under 64 varying lighting conditions are used. We strictly follow the experimental design of [5] for a fair comparison, that is, subjects 2, 5, and 8 of this database are used and the original data are projected onto low-dimensional space (here, LLE [3] method is adopted) before manifold clustering. For the purpose of visualization, we use the class information to label the sample as shown in Figure 4 (a), which will be used as the ground-truth for comparing the different approaches. Note that the class information of the samples are not provided to the clustering methods. We apply mumCluster and the other methods to group the data. As can be seen from Figure 4, our proposed method achieves a better clustering, which has a clustering accuracy of 86.98%, while the clustering accuracy of the other methods are 77.08%, 80.21%, 65.10%, 51.04%, 56.25%, respectively. The total running time of mumCluster on this real-world data is 0.64s, where local intrinsic dimension estimation costs 0.07s while fine clusters identification costs 0.31s.

**Fig. 4.** Clustering results using different methods on a subset of the Yale Face Database B

4.4 The Influence of Parameters

There are three parameters in mumCluster. In this subsection, we examine their impact on the performance of mumCluster by fixing two parameters and varying the concerned

² <http://www.cs.uiuc.edu/homes/dengcai2/Data/FaceData.html>

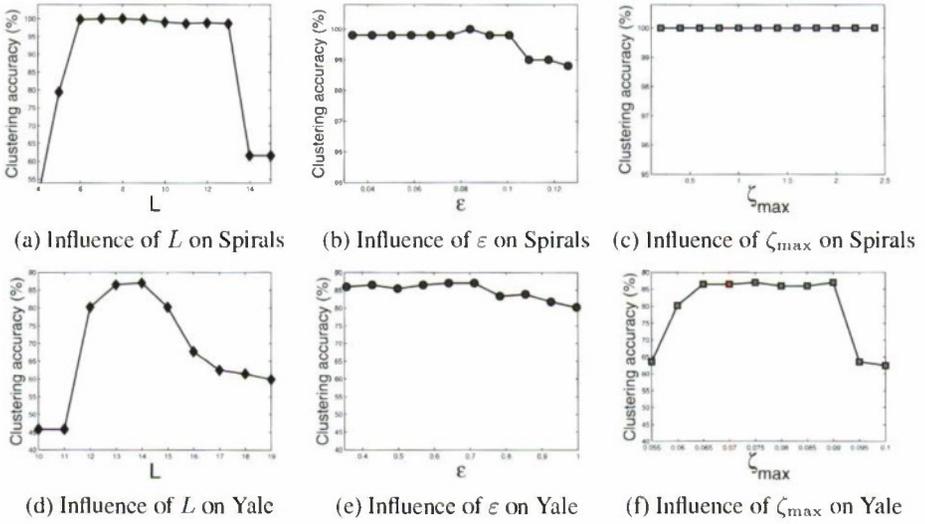


Fig. 5. Influence of parameters on mumCluster

parameter. The results on the spirals data set A and the Yale Face Database B are plotted in Figure 5. We have studied on many other data sets, and the results are similar and thus omitted due to page limit. In general, the optimal values of these parameters depend on the distribution of the samples, while it is easy to see that mumCluster can achieve good performance over a broad rang of these parameters. In detail, the performance of mumCluster is generally insensitive to the setting of L , as long as it is neither too small nor too large. The reason is that L is the number of nearest neighbors which will not capture enough structure information and may lead to many disconnected subgraphs when it is too small, while local property will lose when it is too large. Moreover, as we can see that the results on the Yale data have more fluctuation than on the synthetic data, which show the complexity of the real-world data and thus more attention should be paid to parameter setting. The performance of mumCluster will degenerate when ϵ is large. The reason is that ϵ controls the enlarged area of the intersection points, and it will become too large to ensure a locally linear area. MumCluster is relatively insensitive to the setting of ζ_{max} , as we can see in Figure 5 (c) and (f).

Overall, Figure 5 shows that setting the parameters of mumCluster is not difficult, since the performance of mumCluster is robust to a broad range of parameter values. Moreover, among the three parameters, L has more influence on the performance of mumCluster, which shows that local intrinsic dimension estimation is a key step in our scheme. However, more sophisticated intrinsic dimension estimator can be incorporated into mumCluster to improve the performance, which is our ongoing work.

5 Conclusion

In this paper, we propose a new manifold clustering method, i.e., mumCluster, which can work well when the samples are drawn from hybrid modeling and can adaptively

determine the number of clusters and the intrinsic dimensions. Experimental results show that *mumCluster* is superior to many state-of-the-art manifold clustering methods.

Acknowledgments. The authors are grateful to the referees for their helpful comments. This work was done when Y. Wang was visiting the LAMDA Group, Nanjing University. This work was partially supported by the NSFC (60975038, 60975043), 973 Program (2010CB327903), JiangsuSF (BK2008018) and Jiangsu 333 Program.

References

1. Hartigan, J.A., Wong, M.A.: A K-Means Clustering Algorithm. *Applied Statistics* 28, 100–108 (1979)
2. Seung, H.S., Lee, D.D.: Cognition - the Manifold Ways of Perception. *Science* 290(5500), 2268–2269 (2000)
3. Roweis, S.T., Saul, L.K.: Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science* 290(5500), 2323–2326 (2000)
4. Souvenir, R., Pless, R.: Manifold Clustering. In: *The Tenth IEEE International Conference on Computer Vision*, pp. 648–653 (2005)
5. Vidal, R., Ma, Y., Sastry, S.: Generalized Principal Component Analysis (GPCA). *IEEE Trans. Pattern Anal. Mach. Intell.* 27(12), 1945–1959 (2005)
6. Vidal, R., Tron, R., Hartley, R.: Multiframe Motion Segmentation with Missing Data using Powerfactorization and GPCA. *International Journal on Computer Vision* 79(1), 85–105 (2008)
7. Bradley, P.S., Mangasarian, O.L.: K-plane Clustering. *Journal of Global Optimization* 16(1), 23–32 (2000)
8. Cappelli, R., Maio, D., Maltoni, D.: Multispace KL for Pattern Representation and Classification. *IEEE Trans. Pattern Anal. Mach. Intell.* 23(9), 977–996 (2001)
9. Haralick, R., Harpaz, R.: Linear Manifold Clustering in High Dimensional Spaces by Stochastic Search. *Pattern Recognition* 40(10), 2672–2684 (2007)
10. Shi, J.B., Malik, J.: Normalized Cuts and Image Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* 22(8), 888–905 (2000)
11. Ng, A., Jordan, M., Weiss, Y.: On Spectral Clustering: Analysis and an Algorithm. *Advances in Neural Information Processing Systems* 14, 849–856 (2001)
12. Cao, W.B., Haralick, R.: Nonlinear Manifold Clustering by Dimensionality. In: *The 18th International Conference on Pattern Recognition*, pp. 920–924 (2006)
13. Hastie, T., Tibshirani, R., Friedman, J.: *Elements of Statistical Learning*. Springer, Heidelberg (2001)
14. von Luxburg, U.: A Tutorial on Spectral Clustering. *Statistics and Computing* 17(4), 395–416 (2007)
15. Goldberg, A., Zhu, X., Singh, A., Xu, Z., Nowak, R.: Multi-manifold Semi-supervised Learning. In: *The Twelfth International Conference on Artificial Intelligence and Statistics (AISTATS)*, pp. 169–176 (2009)
16. Fukunaga, K., Olsen, D.R.: Algorithm for Finding Intrinsic Dimensionality of Data. *IEEE Transactions on Computers* c-20(2), 176–183 (1971)
17. Huang, K., Ma, Y., Vidal, R.: Minimum Effective Dimension for Mixtures of Subspaces: a Robust GPCA Algorithm and its Applications. In: *The 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 631–638 (2004)
18. Georghiades, A., Belhumeur, P., Kriegman, D.: From Few to Many: Illumination Cone Models for Face Recognition under Variable Lighting and Pose. *IEEE Trans. Pattern Anal. Mach. Intell.* 23(6), 643–660 (2001)

Exploiting Word Cluster Information for Unsupervised Feature Selection

Qingyao Wu¹, Yunming Ye¹, Michael Ng², Hanjing Su¹, and Joshua Huang³

¹ Harbin Institute of Technology Shenzhen Graduate School, Shenzhen, China
wuqingyao.china@gmail.com, yeyunming@hit.edu.cn

² Department of Mathematics, Hong Kong Baptist University, Kowloon Tong, Hong Kong

³ Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, China

Abstract. This paper presents an approach to integrate word clustering information into the process of unsupervised feature selection. In our scheme, the words in the whole feature space are clustered into groups based on the co-occurrence statistics of words. The resulted word clustering information and the bag-of-word information are combined together to measure the goodness of each word, which is our basic metric for selecting discriminative features. By exploiting word cluster information, we extend three well-known unsupervised feature selection methods and propose three new methods. A series of experiments are performed on three benchmark text data sets (the 20 Newsgroups, Reuters-21578 and CLAS-SIC3). The experimental results have shown that the new unsupervised feature selection methods can select more discriminative features, and in turn improve the clustering performance.

1 Introduction

Feature selection is a process of selecting a feature subspace from the original feature space with some defined criteria. Depending on whether the class label information is required or not, feature selection methods can be classified into two categories, i.e. the supervised approach and the unsupervised approach. The supervised methods rely on the correlation information between features and class label information. The unsupervised methods do not need the class label information, and the goodness of each feature is computed according to its own representation. Complete reviews can be found in [9][13].

The most popular unsupervised feature selection methods usually employ the well-known bag-of-word representation to select feature subspace. In these methods, features are represented with respect to distinct words in the corpus and treated as independent word vector in the vector space model. Feature goodness is affected by the frequency value of the word vector. The higher the frequency, the larger the feature goodness. However, in a high dimensional corpus, there are usually a large portion of low frequency words that are informative to each other. The contribution of these words to document clustering is significant. If we simply select features without considering the correlation between words, there is a big chance that these low frequency words with high discriminative capability will be missed. As a result, the average discriminative capability of the selected feature subspace is decreased.

In this paper, we propose an approach to integrate word clustering information into the process of unsupervised selection methods. In our scheme, we defined a similarity measure of the correlation of co-occurrence between words. After calculating the similarity measure of all pairwise words, we cluster distinct words into groups, and blend the resulted word clustering information with the bag-of-word information to measure the goodness of each feature. This method increases the chance of the inclusion of low frequency features because the defined co-occurrence similarity measure is biased to low frequency words. The basic idea of this approach can be explained intuitively as follows. For example, consider clustering documents about sports into clusters, where each cluster corresponds to individual sport category (e.g., basketball, football, and baseball). The common word “teamwork” related to all three categories may frequently occur in the whole corpus, whereas the discriminative words “dunk” and “layup” which only related to the basketball category may only occur in the documents about basketball. To select feature subspace from the whole feature space in which the discriminative words are less frequent than those common words, we cluster the words into groups. There is a big chance that the discriminative words only occur in the documents about basketball category clustering into a group because they are likely to co-occur together. Furthermore, the word clustering algorithm can sensibly cluster those common words. Thus the new feature selection methods integrating word-cluster information can give more robust estimation to the goodness of the low frequency discriminative words.

Our contributions in this paper are:

- First, we cluster words into groups specifically for the benefit of unsupervised feature selection in this paper. While much study has been devoted to word clustering for text categorization and text clustering [2][3][6][11], but little work has been done on word clustering for unsupervised feature selection. The word clustering has advantages over simple bag-of-word as follows. Word clustering provides a implicitly description to the semantically-related correlations between various words. Word clustering also provides a solution to the sparse and high dimensional challenge of text data set by generating a reduced-size and compact space. But it has to mention that directly using word clusters as features for document clustering, for example in [11][2], will suffer a reduction in performance if the word clusters to compose the feature space are imbalance and impure. Indeed, up-to-date the best results for the well-known Reuters-21578 and 20 Newsgroups data sets are both use words as features [10][12]. As a consequence, it is a natural choice to use the word clustering information in the process of feature selection and select discriminative words to form feature subspace, rather than directly using the word clusters as features to represent documents in the corpus.
- Second, by exploiting word cluster information, we extend three well-known unsupervised feature selection methods and propose three new methods. To compare the text clustering performance with features selected by the new methods and the original methods, we conduct a series of comparative experiments on 3 benchmark data sets, i.e., Reuters-21578, 20 Newsgroups, and CLASSIC3 and the results have shown that the new methods can select better features with high document clustering performance than the original methods.

The rest of this paper is organized as follows. In Section 2 we describe the word clustering algorithm and the similarity measure. Section 3 presents the proposed new unsupervised feature selection methods. Section 4 describes the data sets and the evaluation methods used in our experiments. Section 5 gives a detailed analysis of the experiment results. Finally, we conclude this paper in Section 6.

2 Word Clustering

Data Clustering is a challenging field of data mining research in which its potential applications pose its own special requirements[8]. Clustering is a algorithm to group the data into clusters, so that objects within the same cluster are more similar to objects in other clusters. Often, the clustering performance is influenced by the clustering algorithm and the similarity measure. The choice of clustering algorithm and similarity measure must be suitable for the application target. Without appropriate clustering algorithm and similarity measure, the clustering results can be useless or meaningless.

For word clustering task, there are two typical requirements. First, the text data set is sparse and high-dimensional, the word clustering algorithm should be good at finding clusters in high-dimensional sparse space. Second, the clustering algorithm is required to run efficiently in real-world applications. Some clustering algorithms may work well on handling high-dimensional sparse data set, but they are too time consuming or require users to input certain parameter values, such as the number of clusters. These constraints make them difficult to use.

The *single-linkage* algorithm is an efficient clustering method that can provide a solution to word clustering task. It is a bottom-up agglomerative method that group data into a tree of clusters terminated when the distance between two nearest clusters exceeds a certain threshold. Initially, the *single-linkage* algorithm places each object into individual cluster of its own. The clusters are then merged step-by-step according to some defined similarity measure. Each cluster is represented by all of the objects in the cluster, the similarity between two clusters is measured by the similarity of the closet pair of data objects belonging to two clusters[8]. In clustering the objects, a predetermined minimal similarity threshold is served as the halting criterion.

For word clustering, a measure to compute similarity between words is required. Often, similarities are assessed based on the word vector values in the vector space model. Our raw knowledge about the value of a word is its frequencies in documents. More generally, we can represent the word vector by its frequencies over documents in the training corpus, i.e. $\mathbf{t} = (w(d_1, t), \dots, w(d_{|D|}, t))$, where $w(t, d)$ is the frequency of term t in document d . For computational reason, in what follows, we only consider the presence or absence of a word in the document, that is:

$$w(d, t) = \begin{cases} 1 & t \in d \\ 0 & t \notin d \end{cases} \quad (1)$$

We define the similarity between two word vector as follows:

$$S(\mathbf{t}_i, \mathbf{t}_j) = \min\left(\frac{\mathbf{t}_i \cdot \mathbf{t}_j}{\|\mathbf{t}_i\|_1}, \frac{\mathbf{t}_i \cdot \mathbf{t}_j}{\|\mathbf{t}_j\|_1}\right) \quad (2)$$

This similarity measure is a natural choice for word clustering task for its simplicity and scalability. The result of this similarity measure is in the range of 0 and 1, it is zero just when t_i and t_j are independence. The ratio increases as co-occurrence between two word vectors increases, and bounded by 1. We can thus use the co-occur observation between two words to measure how likely they are to be instances of the same cluster. Since the range of the similarity is in the range between 0 and 1, it is thus more feasible to specify a similarity threshold to determine the termination in the process of clustering. We can set the similarity threshold to be 0.5. Since the purpose of the word clustering algorithm is to provide an implicitly additional correlation information between various words, the performance of feature selection is not sensitive to the similarity threshold.

3 Word Clusters for Unsupervised Feature Selection

After clustering all words in the corpus into clusters, an additional step to exploit the word cluster information is added before selecting features. We then blend the word cluster information with the bag-of-word information to measure the goodness of individual features. With this hybrid method, we extend three well-known unsupervised feature selection methods, i.e. Document Frequency, Term Contribution and Term Quality, and proposed three new methods, called *word-cluster* approach, in which both word and word clusters information are included.

3.1 Word Clusters for Document Frequency (DF and wc_DF)

Document frequency (DF) as a feature selection criterion for a term t can be described as follows, $DF(t) = |D_t|$, where $|D_t|$ is the number of documents in which term t occurs. The higher the document frequency, the better the feature.

The DF feature selection method can be used for both supervised document categorization [13] and unsupervised document clustering [9]. This method assume that the contribution of low frequency words is insignificant. Improvement in performance is also possible if low frequency terms happened to be noises. However, low frequency words may contain useful discriminative information in clustering data in the domain of high dimensional with many classes. This idea is consistent with the popular inverse document frequency weighting scheme in the area of information retrieval.

To extend the DF feature selection, we take the word cluster size into account in the new word-cluster DF method. Formally, the word-cluster DF criterion for term t can be defined as follows:

$$wc_DF(t) = |D_t| \times (1 + \log |C_t|) \quad (3)$$

where $|C_t|$ is the size of cluster in which term t is included. Here, $|D_t|$ refer to the importance in document aspect, and $|C_t|$ refer to the importance in word aspect. This method indicates that the importance of a term t can be improved by increasing the number of documents containing the term or the number of words correlated with the term.

3.2 Word Clusters for Term Contribution (TC and wc_TC)

Term contribution is proposed by Liu et al.[9]. It is a criterion to measure the contribution of a term to discriminate documents in the data set. In this method, the contribution of a term is equivalent to its contribution to all pairwise document similarity in the corpus. Formally, the contribution of a term t to the similarity of a pairwise document d_i and d_j can be defined as follows:

$$TC(t, d_i, d_j) = w(t, d_i) \times w(t, d_j) \quad (4)$$

The contribution of term t to all pairwise document in the corpus is defined as follows:

$$TC(t) = \sum_{i,j \cap i \neq j} w(t, d_i) \times w(t, d_j) \quad (5)$$

where $w(t, d_i)$ and $w(t, d_j)$ is the weight value of term t in document d_i and d_j , respectively. Often, the *tf-idf* weight value is used. Formally, the *tf-idf* value of term t in document d can be computed as follows:

$$w(t, d) = \frac{tf_{td}}{\sum_t tf_{td}} \log \frac{N}{|D_t|} \quad (6)$$

where tf_{td} is the term frequency of term t in document d , $\sum_t tf_{td}$ is the sum of all term frequencies in document d , it is used to normalize tf_{td} to prevent a bias towards longer documents, N is the total number of documents in the corpus, and $|D_t|$ is the number of documents in which term t occurs, here $N/|D_t|$ refer to be the inverse document frequency of term t .

To extend the TC feature selection method, we propose a new *tf-idf* type weighting scheme in which both word and word cluster information are included. Formally, the word-cluster *tf-idf* value of a term t can be defined as follows:

$$w'(t, d) = \frac{tf_{td}}{\sum_t tf_{td}} \times \log \frac{N}{|D_t|} \times (1 + \log |C_t|) \quad (7)$$

where $|C_t|$ is the size of cluster in which term t is included. High value in the new weight scheme corresponding to high term frequency, low document frequency and significant word cluster in the corpus.

The word-cluster TC criterion is then given by straightforward applying the new *tf-idf* weight scheme to the original word-solely TC criterion. It is given by:

$$wc_TC(t) = \sum_{i,j \cap i \neq j} w'(t, d_i) \times w'(t, d_j) \quad (8)$$

3.3 Word Clusters for Term Quality(Q_0 , Q_1 and wc_Q)

Term quality is proposed and evaluated by Dhillon[5]. It is a criterion to measure the goodness of a target term via its distribution. If we consider two major variables, t and

D with respect to the target term and the documents in the corpus. Our knowledge to the correlation between them is the term frequency tf_{td} of occurrence in pairs (t, d) in the corpus. With this knowledge, we define the distribution of a term t over documents D to be $p_t(d) = tf_{td}$, where tf_{td} is the term frequency of term t in document d , we refer to this distribution as *document distribution*.

To evaluate the feature goodness according to its *document distribution*. Dhillon defined the distribution variance as a measure of the discriminative capability of a distribution. The variance of *document distribution* p_t over documents in the corpus is given by:

$$Var(p_t, D) = E[p_t^2] - E[p_t]^2 = \frac{\sum_{d \in D} tf_{td}^2}{|D|} - \frac{1}{|D|^2} \left[\sum_{d \in D} tf_{td} \right]^2 \quad (9)$$

where $E[p_t]$ is the expected value (mean) of distribution p_t .

Dhillon applied the variance formula to feature selection and defined a similar criteria call term quality Q_0 :

$$Q_0(t) = |D| \times Var(p_t, D) = \sum_{d \in D} tf_{td}^2 - \frac{1}{|D|} \left[\sum_{d \in D} tf_{td} \right]^2 \quad (10)$$

Q_0 criteria is influenced by the dispersion of all documents in the corpus. The larger the dispersion, the better the term discriminate capability. However, it has poor performance while applying to sparse data set in which a large portion of low frequency terms only occur in documents related to a particular category. There is a big chance these low frequency terms are missed while using Q_0 to select feature subspace. To remedy this, Dhillon introduced another term quality, called Q_1 , which is influenced by the dispersion of the documents that contain the target term at least once. Formally, it is given by:

$$Q_1(t) = |D'| \times Var(p_t, D) = \sum_{d \in D'} tf_{td}^2 - \frac{1}{|D'|} \left[\sum_{d \in D'} tf_{td} \right]^2 \quad (11)$$

where D' is the document set in which term t occurs at least once. A major difference between Q_0 and Q_1 is that Q_0 measure the target term through the aspect of total documents, whereas Q_1 only considers the documents in which the target term occurs.

To extend the term quality feature selection method, we define a new distribution for the target term that is the document frequency value on its word cluster. Here we consider two variables, the target term t and its word cluster C_t . Our knowledge to the correlation between them is the document frequency df_w of word w , and w is in the word cluster in which term t included. Thus the distribution of a term t over word cluster C_t is then given by $p_t(w) = df_w$, we refer to this distribution as *word distribution*.

The distribution variance p_t over words in the cluster C_t is given by:

$$Var(p_t, C_t) = E[p_t^2] - E[p_t]^2 = \frac{\sum_{w \in C_t} df_w^2}{|C_t|} - \frac{1}{|C_t|^2} \left[\sum_{w \in C_t} df_w \right]^2 \quad (12)$$

The word-cluster term quality criterion is then given by putting the *document distribution* variance and *word distribution* variance into together as follows:

$$wc_Q(t) = Var(p_t, D) \times Var(p_t, C_t) \quad (13)$$

The combination of the document variance and word variance provides additional word-cluster knowledge and thus makes the new method less sensitive to term frequency.

4 Experimental Results

To compare the document clustering performances with respect to features selected by the new methods and features selected by the original methods, we conduct a series of comparison experiments on three public benchmark text data sets, i.e., 20 Newsgroups, Reuters-21578 and CLASSIC3.

4.1 Data Sets

The CLASSIC3 data set [7] is available on the SMART system from Cornell's Web site ¹. It consists of 33,242 features and 3,896 document abstracts, and contains three categories, i.e., MEDLINE, CISI, and CRANFIELD, from different specific domains. MEDLINE consists of 1,033 abstracts from medical papers, CISI consists of 1,460 abstracts from information retrieval papers, and CRANFIELD consists of 1,400 abstracts from aeronautical systems papers. The characteristics between these categories are very different. The overlapping between keywords of different categories is not large. It is thus not difficult to cluster the CLASSIC3 corpus.

The Reuters-21578 is a corpus for text mining contains 21,578 new stories appeared in the Reuters newswire in 1987². We used the modified Apte ("ModeApte") split contains 9,603 training documents and 3,299 test documents. But we discarded those documents have no labels, and the remained data set consists of 7,063 training documents and 2,742 test documents. Furthermore, we generated our Reuters subset by selecting the largest 10 categories which have maximum positive training documents[4], and then discarded those documents belong to more than one category. The resulted Reuters subset contains 19,206 features for 5,973 documents. It is note that the number of documents in different categories are very different. The largest category contains 2,698 documents, whereas the smallest category only contains 80 documents. The purpose of generating this data set is to evaluate the performance on corpus in domain of many imbalance classes.

The 20 Newsgroups corpus contains 19,997 documents from the Usenet newsgroups collection³. In the experiment, we used a benchmark minisubset of 20 Newsgroups corpus that provide by UCI machine learning archive [1]. The 20 Newsgroups subset

¹ CLASSIC3 can be found at: <ftp://ftp.cs.cornell.edu/pub/smart>

² Reuters-21578 can be found at:

<http://www.daviddlewis.com/resources/testcollections/reuters21578/>

³ The 20 Newsgroups can be found at:

<http://kdd.ics.uci.edu/databases/20newsgroups/20newsgroups.html>

consists of 2,000 news documents and 26,620 features. It contains 20 categories, each of which has 100 documents. Most of the document is designated into one category, but the categories in the corpus are semantically close, such as “comp.sys.imb.pc.hardware” and “comp.sys.mac.hardware”, “comp.graphics” and “comp.windows.x”. The similarity and overlapping between different categories makes it difficult to correctly group the 20 Newsgroups corpus into clusters. The purpose of choosing this data set is to evaluate the performance on corpus in domain of many similar classes.

These three corpus are frequently used as benchmark data set in the task of text mining. They represented considerable diversity of number of classes, data in size, data imbalance and data similarity. We preprocess these corpus using the *DRAGON* toolkit [14]. The detail of these three corpus are list in Table 1.

Table 1. Data sets used in the experiments

Data set	CLASSIC3		Reuters		20NG	
# Variable	33,242		19,206		26,620	
# Sample	3,896		5,973		2,000	
# Class	3		10		20	
Class	Name	# Sample	Name	# Sample	Name	# Sample
C1	MEDLINE	1,033	Earn	2,698	1	100
C2	CISI	1,456	Acq	1,471	2	100
C3	CRANFIELD	1,400	Money-fx	401	3	100
C4			Grain	334	4	100
C5			Crude	295	5	100
C6			Trade	292	6	100
C7			Interest	169	7	100
C8			Ship	134	8	100
C9			Money-supply	99	9	100
C10			sugar	80	10	100
...					...	100
C20					20	100

4.2 Evaluation Measures

We use two quality measures to evaluate the effectiveness of the selected features for text clustering, i.e., Entropy(E), and F-measure(F). The first measure Entropy provides a measure of the purity of a cluster. The cluster contains a large portion of objects from different classes has a large entropy. The smaller the entropy, the better the performance. The second measure F-Measure is a common used performance evaluation in information retrieval. It combines the effect of precision and recall. The higher the F-measure, the better the clustering result.

For the entropy evaluation measure, we denote $C = \{C_1, C_2, \dots, C_k\}$ as the obtained clusters and $C^* = \{C_1^*, C_2^*, \dots, C_{k'}^*\}$ as the correct classes, k and k' respectively to their cluster number. Let $|C_i|$ be the number of documents in i th obtained cluster and $|C_i^*|$ be the number of documents in i th corrected class. Given $C_i \in C$, the the entropy of a target cluster C_i is defined to be:

$$E_i = -\frac{1}{\log k'} \sum_{j=1}^{k'} p_{ij} \log(p_{ij}) \quad (14)$$

where p_{ij} is the probability that the target cluster C_i belongs to the correct class C_j^* . Formally, p_{ij} is given by:

$$p_{ij} = \frac{|C_i \cap C_j^*|}{|C_i|} \quad (15)$$

where $|C_i \cap C_j^*|$ is the number of documents of the class C_j^* that are assigned to cluster C_i , that is, the overlap between C_i and C_j^* . Given $|C|$ as the total number of documents in the corpus, the total entropy for the target clusters is computed as follows:

$$E = \sum_{i=1}^k \frac{|C_i|}{|C|} E_i \quad (16)$$

Given $C_i \in C$ and $C_j^* \in C^*$, the precision, recall and F-Measure of the target cluster C_i with respect to class C_j^* is defined to be: $P(i, j) = |C_i \cap C_j^*|/|C_i|$, $R(i, j) = |C_i \cap C_j^*|/|C_j^*|$ and $F(i, j) = 2P(i, j) \cdot R(i, j)/(P(i, j) + R(i, j))$.

For a particular target cluster, we choose the class that shares with most documents with the target cluster as the correct class to evaluate its performance. That is, $F_i = \max\{F(i, j) | j = 1, \dots, k\}$. The total F-Measure for all clusters is defined as follows:

$$F = \sum_{i=1}^k \frac{|C_i|}{|C|} F_i \quad (17)$$

4.3 Comparison Experiments

To compare the document clustering performance with respect to feature subspace selected by the word-cluster methods and feature subspace selected by the word-solely methods, we conduct a series comparison experiments on the CLASSIC3, Reuters-21578, and 20 Newsgroups corpus. In these experiments, we used the *single-linkage* algorithm and similarity measure introduced in Section 2 to group words into clusters, and the default similarity threshold is set to be 0.5. For documents clustering, we used the group-average agglomerative method as document clustering algorithm. The distance between two clusters is measured by the average cosine distance between documents with respect to two clusters.

We carried out the document clustering algorithm on three corpus. For each corpus, the document clustering algorithm was executed in different percentage of features from 2 to 40. For the same percentage, we carried out the document clustering algorithm with features selected by each method, and computed their F-Measure(F) and Entropy(E) results. The results we reported are averaged over the 3 folds cross validation. Fig. 1, Fig. 2 and Fig. 3 show the F-Measure and Entropy comparison results. In the figures, the left labels correspond to the F-Measure(F) scale, and the right labels correspond to the Entropy(E) scale. For the F-Measure results, the solid triangle plots are the F-Measure score of the word-cluster methods, and the solid square plots are the F-Measure score of the word-solely methods. For the entropy results, the hollow triangle plots refer to

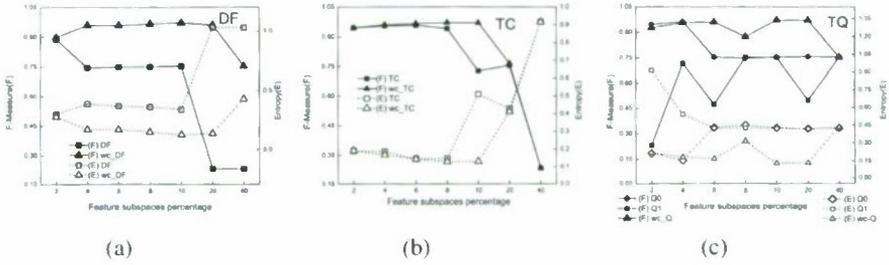


Fig. 1. Comparison on CLASSIC3 corpus with different feature subspaces, (a)(b)(c) are methods with respect to document frequency, term contribution and term quality

the entropy score of the word-cluster methods, and the hollow square plots refer to the entropy score of the word-solely methods.

Figure 1 shows the plots of the F-Measure and Entropy results on the CLASSIC3 corpus with different feature subspaces separately selected by the word-cluster methods and the word-solely methods. We can clearly see that the word-cluster methods outperform the word-solely methods in both F-Measure and Entropy in all feature subspaces, i.e., word-cluster methods attained higher F-Measure and lower Entropy results. Specifically, for the *document frequency* type methods showed in Fig. 1(a), we can observe that the word-cluster method significantly outperform the word-solely method, the F-Measure of the word-cluster method *wc_DF* is almost 20% larger than the F-Measure of the word-solely method *DF*. For the *term contribution* type methods showed in Fig. 1(b), we can see that the performance of the word-cluster method and the word-solely method is comparable, but the word-cluster method *wc_TC* is better on 8% feature subspace and 10% feature subspace. Another observation is that the performance decrease rapidly after selecting 20% feature subspace. For the *term quality* type methods showed in Fig. 1(c), we can see that the square plots are below the triangle plots for F-Measure result comparison in all feature subspaces.

On the whole, the performance of the word-cluster methods and the performance of the word-solely methods are comparative when the percentage of feature subspace is less than 6%. As the feature percentage increase, we can observe that the word-cluster methods are more stable, because the curves of the word-cluster methods are smooth, while the curves of the word-solely methods are uneven.

Figure 2 shows the plots of the F-Measure and Entropy results on the Reuters-21578 corpus with different feature subspaces separately selected by the word-cluster methods and the word-solely methods. We can see that the average performance on this corpus is worse than those on the CLASSIC3 corpus. The decrease of performance on the Reuter-21578 corpus may due to the imbalance property of this corpus. But we can clear see that the word-cluster methods outperform the word-solely methods on small feature subspaces that are less than 10%. This result indicates that the word-cluster methods are especially effective while only selecting a small feature subspace. Another observation is that the performance of the word-cluster methods are similar to those of the word-solely methods on large feature subspace. In fact, the word-cluster methods slightly outperform on most of the percentage. When the selected feature subspace is

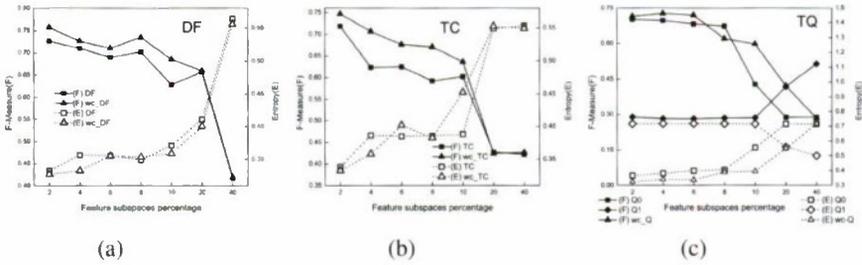


Fig. 2. Comparison on Reuters-21578 corpus with different feature subspaces, (a)(b)(c) are methods with respect to document frequency, term contribution and term quality

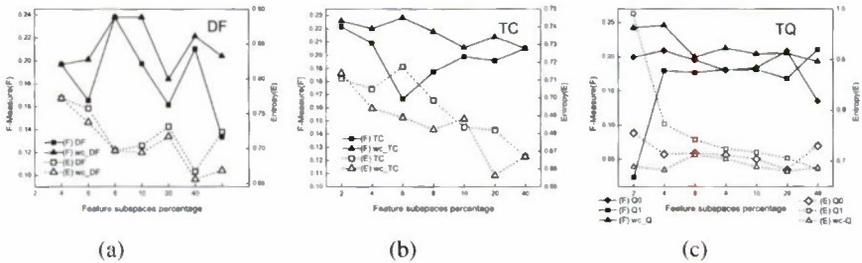


Fig. 3. Comparison on 20 Newsgroup corpus with different feature subspaces, (a)(b)(c) are methods with respect to document frequency, term contribution and term quality

large, both word-cluster method and word-solely method have a big chance to select feature subspace in which no informative features are included, and thus the document clustering performance is reduced.

Figure 3 shows the plots of the F-Measure and Entropy results on the 20 Newsgroups corpus with different feature subspaces separately selected by the word-cluster methods and the word-solely methods. The overall performance on this corpus is comparatively worse than those performance on other two corpus, because the categories in the 20 Newsgroups are semantic similar and overlapped to each other. However, the improvement of the word-cluster methods is significant in this corpus. We can see that the word-cluster methods clearly outperform the word-solely methods in almost all feature subspaces. This result indicates that the word-cluster methods is especially effective for complex corpus in the domain of many classes and high overlapping.

In summary, the experiment results show that the word-cluster methods outperform the word-solely methods in most of feature subspace. These results indicate that the word-cluster methods could select more discriminative features, and thus the document clustering performance is improved.

5 Conclusions

In this paper, we define a clustering algorithm and a similarity measure to group words in the corpus into clusters, and blend the word cluster information with the bag-of-word

information to select feature subspace for document clustering. In this way, we extend three well-known unsupervised selection methods and proposed three new methods. We have conducted a series of comparison experiments on three benchmark corpus, and the results show that the document clustering performance on feature subspaces selected by the word-cluster methods outperform those selected by the word-solely methods.

Acknowledgements

This research is supported in part by NSFC under Grant no. 60603066, China National High-tech Program under Grant no. 2007AA01Z436, and Shenzhen Science and Technology Program under Grant nos. NSKJ-200707, 08CXY-44.

References

1. Asuncion, A., Newman, D.J.: UCI machine learning repository (2007)
2. Bekkerman, R., El-Yaniv, R., Tishby, N., Winter, Y.: Distributional word clusters vs. words for text categorization. *The Journal of Machine Learning Research* 3, 1183–1208 (2003)
3. Dai, W., Xue, G.R., Yang, Q., Yu, Y.: Co-clustering based classification for out-of-domain documents. In: *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 210–219. ACM, New York (2007)
4. Debole, F., Sebastiani, F.: An analysis of the relative hardness of Reuters-21578 subsets. *Journal of the American Society for Information Science and technology* 56(6) (2005)
5. Dhillon, I.S., Kogan, J., Nicholas, C.: Feature selection and document clustering. *A Comprehensive Survey of Text Mining*, 73–100 (2003)
6. Dhillon, I.S., Mallela, S., Kumar, R.: A divisive information theoretic feature clustering algorithm for text classification. *The Journal of Machine Learning Research* 3, 1287 (2003)
7. Dhillon, I.S., Mallela, S., Modha, D.S.: Information-theoretic co-clustering. In: *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, NY, USA, pp. 89–98 (2003)
8. Han, J., Kamber, M.: *Data mining: concepts and techniques*. Morgan Kaufmann, San Francisco (2006)
9. Liu, T., Liu, S., Chen, Z., Ma, W.: An evaluation on feature selection for text clustering. In: *Proceeding of the 20th ICML International Conference on Machine Learning*, vol. 20, p. 488 (2003)
10. Schapire, R.E., Singer, Y.: BoosTexter: A boosting-based system for text categorization. *Machine learning* 39(2), 135–168 (2000)
11. Slonim, N., Tishby, N.: Document clustering using word clusters via the information bottleneck method. In: *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 208–215. ACM, New York (2000)
12. Weiss, S.M., Apte, C., Damerau, F.J., Johnson, D.E., Oles, F.J., Goetz, T., Hampp, T.: Maximizing text-mining performance. *IEEE Intelligent Systems and their Applications* 14(4), 63–69
13. Yang, Y., Pedersen, J.O.: A comparative study on feature selection in text categorization. In: *Proceedings of the 14th ICML International Conference on Machine Learning*, pp. 412–420 (1997)
14. Zamir, O., Etzioni, O.: Dragon Toolkit: Incorporating Auto-learned Semantic Knowledge into Large-Scale Text Retrieval and Mining. In: *Proceedings of the 19th IEEE International Conference on Tools with Artificial Intelligence*, Patras, Greece (2007)

Sparse Representation: Extract Adaptive Neighborhood for Multilabel Classification

Shuo Xiang, Songcan Chen, and Lishan Qiao

Department of Computer Science and Engineering, Nanjing University of Aeronautics and Astronautics, China

Abstract. Unlike traditional classification tasks, multilabel classification allows a sample to associate with more than one label. This generalization naturally arises the difficulty in classification. Similar to the single label classification task, neighborhood-based algorithms relying on the nearest neighbor have attracted lots of attention and some of them show positive results. In this paper, we propose an Adaptive Neighborhood algorithm for multilabel classification. Constructing an adaptive neighborhood is challenging because specified information about the neighborhood, e.g. similarity measurement, should be determined automatically during construction rather than provided by the user beforehand. Few literature has covered this topic and we address this difficulty by solving an optimization problem based on the theory of sparse representation. Taking advantage of the extracted adaptive neighborhood, classification can be readily done using weighted sum of labels of training data. Extensive experiments show our proposed method outperforms the state-of-the-art.

1 Introduction

Multilabel classification has been a popular issue in pattern recognition & machine learning and is encountered in a variety of application domains. For instance, in biology, a gene or protein may possess several functionalities and in natural scene classification, a picture of the beach may also include boats, trees and even a city as its contents. Behind these appearances lies the fact that one object is allowed to associate with more than one labels. Solving classification tasks of multilabel scenario is naturally a generalization of traditional task and possesses much more practical value as well as difficulties.

Several methods taking advantage of traditional classification algorithm, e.g. AdaBoost, SVM, EM, have been proposed to solve this problem. Recent research [1,2] shows that neighborhood-based algorithms relying on the nearest neighbor can achieve good results in multilabel classification task, just like in case of single label. However, the way of choosing neighborhood in these works is based on *K Nearest Neighbor (KNN)*, in which several parameters should be given in advance such as the similarity measurement and the size of the neighborhood K . Constructing an adaptive neighborhood that can get rid of these specifications would be helpful but challenging. In this paper, we address the difficulty by extracting this adaptive neighborhood with an optimization problem based on the theory of sparse representation and further use it for multilabel classification. To our best knowledge, we are not aware of any similar work using this technique to handle the multilabel classification problem.

The rest of the paper is organized as follows. In the next section we give a brief review of previous work on the topic of multilabel classification and sparse representation. Then we present our *Adaptive Neighborhood(AN)* algorithm and report the experimental results. Finally we conclude this paper and point out some promising work in the future.

2 Related Work

2.1 Multilabel Classification

Multilabel classification began to be widely concerned due to the work of Schapire and Singer [3]. They presented a boosting-based system BoosTcXter for text categorization and also provided several useful measurements that can be extended to other multilabel classification tasks. Besides, they pointed out that controlling the complexity of the overall learning system is an important research issue. To control this complexity while having a small empirical error, Elisseeff and Weston proposed the RankSVM [4] method. As in Support Vector Machine(SVM), a linear model is defined so as to minimize the empirical error measured by the ranking loss and control the complexity of the resulting model simultaneously.

Zhang and Zhou introduced a lazy way of multilabel classification named *ML-KNN* [1]. In their algorithm, *K Nearest Neighbor* in the training set is first computed for an unseen sample, then a *Maximum A Posteriori(MAP)* method is taken to perform the classification, based on the statistical information gained from the label sets of neighbor instances. Motivated by this lazy way method, Cheng and Hullermeier gave *IBLR-ML* algorithm [2] which combines the instance-based learning and logistic regression and allows one to capture the interdependencies between the class labels in a proper way. Experiments on public data sets show that, among several existing multilabel classification algorithm, both *ML-KNN* and *IBLR-ML* show not only positive results but also achieve the state-of-the-art classification performance. However, both of these methods are based on the *KNN* which can easily fall into the predicament of suitable similarity measurements and the size of the neighborhood.

2.2 Sparse Representation

Theory of Sparse Representation is closely related to our work. It has been quite popular in machine learning area, including face recognition [5], dimensionality reduction [6], image super-resolution [7] and image denoising [8]. Sparse solution of underdetermined systems of linear equations lies at the heart of this theory. As stated in [9], finding such solution can be formulated as the following optimization problem(P_0):

$$\begin{aligned} \min_w \quad & \|w\|_0 \\ \text{s.t.} \quad & z = Xw \end{aligned} \quad (1)$$

Unfortunately, although l_0 -norm is a straightforward measurement of sparsity, problem P_0 has been proved to be NP-hard [10]. To overcome this prohibitive computation issue, a compromising way is to deal with P_1 instead:

$$\begin{aligned} \min_w \quad & \|w\|_1 \\ \text{s.t.} \quad & z = Xw \end{aligned} \quad (2)$$

which is a convex optimization and can be readily solved by linear programming [11]. P_1 is the central focus of sparse representation and has been shown to have exactly the same solution as P_0 when the solution is very sparse. [9]

Sparse representation has been involved in many classification tasks, one of which belongs to Wright's work [5] on robust face recognition. According to their paper, samples from the same class are modeled as lying on a linear subspace. Given sufficient training samples of the i th class, $X_i = [x_{i,1}, \dots, x_{i,n_i}] \in R^{d \times n_i}$, any test sample $z \in R^d$ from the same class would be able to be approximately written as the linear combination of training samples associated with the i th class:

$$z = w_1 x_{i,1} + \dots + w_{n_i} x_{i,n_i} = X_i w$$

Following the idea above, for any unseen sample, finding a sparse representation in all the training samples would typically yield the solution with nonnegative entries associated with training examples of the same class, as shown in the following results, from which we can see that sparse representation is able to capture the discriminant nature behind the samples:

$$z = Xw = [X_1, \dots, X_c][0, \dots, 0, w_1, \dots, w_{n_i}, 0, \dots, 0]^T$$

The sparse representation can be obtained by solving P_1 . In realistic tasks, the exact representation of test sample may not be able to be achieved due to noise. Usually a stable version is considered instead:

$$\begin{aligned} \min_w \quad & \|w\|_1 \\ \text{s.t.} \quad & \|z - Xw\|_2 < \epsilon \end{aligned} \quad (3)$$

where ϵ is an error tolerance. This is a convex programming and can be efficiently solved. With the obtained representation, prediction of a test sample is able to be made by choosing the class with least residual. The algorithm achieves positive results on several public data sets with high accuracy and robustness to occlusion.

3 Adaptive Neighbor(AN) Algorithm

3.1 Problem Setting

Consider the following multilabel classification with:

training set: $Tr = \{(x_i, Y_i)\}_{i=1}^n, (x_i \in \mathcal{X}, y_i \in \mathcal{Y})$

test set: $Te = \{z_i\}_{i=1}^m, (z_i \in \mathcal{X})$

Our goal is to learn a classifier:

$$f: \mathcal{X} \times \mathcal{Y} \mapsto \mathcal{R}$$

which tends to assign higher value to (z, y_i) if y_i belongs to Y_z . From f we can easily predict the label of an unseen sample, e.g. $predict(z, y_i) = \llbracket f(x, y_i) \geq \theta \rrbracket$, θ is a threshold. Another statistic we would like to gain is the rank information between different labels, the function $rank_f(z, y_i)$ ranks different labels according to the corresponding value of $f(z, y_i)$, where higher value of f gets lower(better) rank position.

3.2 Our Method

Extensively applied in different machine learning tasks, ranging from single label classification to dimensionality reduction [12,13] and multilabel classification [1], *KNN* usually serves as an intermediate step to seek the connections between samples. However, neighborhood information gained from *KNN*, largely based on the choice of similarity measurement and the size of the neighborhood, presents a simple but limited portrait of the correlations between samples.

In order to capture the discriminant nature behind the data, our work focuses on designing an effective construction of an adaptive neighborhood on which multilabel classification task can be efficiently carried out. By adaptive, we mean, this neighborhood is determined by the natural structure behind the data and we don't have to prescribe the parameter like the number of neighbors K or a specific way of similarity measurement. Motivated by sparse representation in face recognition [5], we summarize this procedure in a similar optimization problem(P_{AN}):

$$\begin{aligned} \min_w \quad & \|z - Xw\|_2^2 + \lambda \|w\|_1 \\ \text{s.t.} \quad & w \geq 0 \end{aligned} \tag{4}$$

X is a d by n matrix whose columns contain the training data of dimension d . z is a single test sample and our goal is to seek the sparsest coefficient w while keeping the residual as small as possible. This formulation is able to capture exactly the same kind discriminant nature as sparse representation stated in the previous section. However, our method still differs from sparse representation in the objective function and the constraint as follows:

- Different from sparse representation which aims at finding a sparse solution with best reconstruction results, our method concerns more to find out the information of neighborhood in which the nonnegativity is necessary.
- The nonnegativity constraint can provide us a straightforward interpretation of the relation between the test sample and the training sample, where larger value of w_j means that the i th training sample is "more similar" to the test sample z and vice versa.

Based on the facts above, we claim that an adaptive neighborhood for each test sample is obtained by noticing that we don't need to prescribe any concrete way of similarity measurement between samples or the size of the neighborhood. Unlike sparse representation's choosing class with least residual in classification [5], we design the classifier in a simpler weighted sum way: for a label $l \in L$ and a given test sample z , $f(z, y_l) = \sum_j w_j * Y_{lj}$, Y contains the true label of training data, each in a column. Algorithm 1 shows the the complete description.

3.3 Comparison with Previous Work

Compared to previous the state-of-the-art works like *ML-KNN* and *IBLR-ML*, several remarkable differences should be emphasized for our method which makes multilabel classification done effectively and efficiently.

First, the neighbors chosen by our algorithm is generally different from that of *KNN*. Inherited from sparse representation, *AN* tends to select those neighbors that share the same underlying subspace, as can be seen from Figure 1.

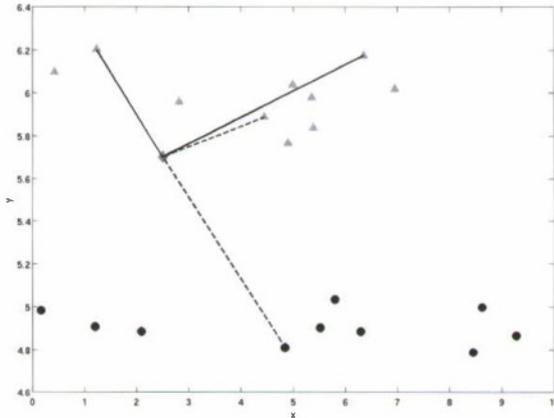


Fig. 1. Data from two affine subspace ($y = 5.0, y = 6.0$) with gaussian noise added. The solid line shows the neighbors selected by *AN* and the dashed line gives that of *2-Nearest Neighbors (2NN)*. *2NN* selects neighbors with least distances while the neighbors chosen by *AN* automatically (with the size of two coincidentally) tend to lie on the same subspace which are much more discriminative [5].

Second, we don't need to prescribe the size of the neighborhood K as in *ML-KNN* and *IBLR-ML*. K is set to the number of nonnegative elements in w which is naturally obtained from the above optimization. Although we can still fix the value of K by choosing the K largest elements of w , it is advisable that different samples would belong to the different neighborhoods which have different sizes.

In addition, due to the natural discriminant property of sparsity, no further complicated classifier is required, a simple weighted sum would suffice. This makes the classification procedure more efficient.

4 Experiments

In this section, experiments are conducted on public multilabel classification data sets, which serve both to demonstrate the efficacy of the proposed method and to validate the claim we have made in the previous sections. We compare our results with the state-of-the-art, including *ML-KNN* and *IBLR-ML*, of which the implementations are provided

Algorithm 1. Adaptive Neighborhood

Input:

- X : training data
- Y : training label set
- z : test sample
- θ : threshold, λ : regularizer

Output:

- f : classifier
- $predict$: predicting function

Procedure:

for all test sample z **do**

Solve the optimization Problem:

$$\min_w \quad \|z - Xw\|_2^2 + \lambda\|w\|_1 \quad s.t. \quad w \geq 0$$

Normalize w

$$f(z, \cdot) = Yw$$

for $j = 1$ **to** $|L|$ **do**

if $f(z, y_j) \geq \theta$ **then**

$$\quad \quad \quad predict(z, y_j) = 1$$

else

$$\quad \quad \quad predict(z, y_j) = -1$$

end if

end for

end for

by their original authors. Our algorithm can be efficiently implemented using the sparse learning package `ll_ls`¹ or SLEP [14].

4.1 Measurement

Unlike traditional loss function of single label classification, special criterion should be considered while evaluating the performance of multilabel task. Here we utilize the measurements that provided in [3].

- Hamming Loss:

$$hloss(predict, x, Y) = \frac{1}{n} \sum_{i=1}^n \frac{1}{|L|} |predict(x_i) \oplus Y|$$

- OneError:

$$OneError(f, x, Y) = \frac{1}{n} \sum_{i=1}^n \llbracket \arg \max_y f(x_i, y) \notin Y_i \rrbracket$$

¹ http://www.stanford.edu/boyd/ll_ls/

– Coverage:

$$\text{Coverage}(f, x, Y) = \frac{1}{n} \sum_{i=1}^n \max_{y \in Y_i} \text{rank}_f(x_i, y) - 1$$

– Ranking Loss:

$$\text{rloss}(f, x, Y) = \frac{1}{n} \sum_{i=1}^n \frac{1}{|Y_i| |\bar{Y}_i|} \times \\ |\{(y_1, y_2) | f(x_i, y_1) \leq f(x_i, y_2), (y_1, y_2) \in Y_i \times \bar{Y}_i\}|$$

– Average Precision:

$$\text{AvgPrec}(f, x, Y) = \frac{1}{n} \sum_{i=1}^n \frac{1}{|Y_i|} \times \\ \sum_{y \in Y_i} \frac{|\{y' | \text{rank}_f(x_i, y') \leq \text{rank}_f(x_i, y), y' \in Y_i\}|}{\text{rank}_f(x_i, y)}$$

The operator \oplus in *Hamming Loss* means symmetric difference which measures the number of labels that we have misclassified during the test phase. In *One Error*, function $[\cdot]$ takes 1 if the parameter it takes holds true and the whole statistic calculates the times the label we classified with most confidence is actually incorrect. *Coverage* measures how far we need to go down the label list to cover all the positive label and *Ranking Loss* provides the average fraction of pairs that are not correctly ordered. Similar to the concept of precision in Information Retrieval, *Average Precision* gives the mean precision on every label.

Table 1. Statistics of the data sets used in the experiments

DATA SET	INSTANCE	ATTRIBUTE	LABEL
<i>genbase</i>	662	1186	27
<i>medical</i>	978	1449	45
<i>enron</i>	1702	1001	53
<i>bibtex</i>	7358	1836	159

4.2 Data Sets

Four data sets¹: *genbase*, *medical*, *enron* and *bibtex* are chosen for our experiments. Data set *genbase* is derived from the task of protein classification [15], where each protein can associate with at most 27 labels. The data set contain 662 instances of 1185 dimensions. 978 instances of dimension 1449 each with 45 labels are contained in the data set of *medical*. It comes from the international challenge of classifying clinical free text using natural language processing, which aims to create and train computational intelligence algorithms that automate the assignment of *ICD-9-CM* codes to clinical

¹ <http://mlkd.csd.auth.gr/multilabel.html>

free text. Data set *enron* is derived from the *UC Berkeley Enron Email Analysis Project* and contains Email data from about 150 users, mostly senior management of Enron. After processing, the current data set is comprised of 1702 instances with the dimension of 1001 and 53 labels are involved. The last data set we use is relative large. *bibtex* was used to solve the automated tag suggestion problem [16], containing 7395 instances of 1836 dimension with 159 labels. An overview of all the data is provided in Table 1.

4.3 Parameter Setting

As pointed in the previous section, K Nearest Neighbors are involved in both algorithms of *ML-KNN* and *IBLR-ML*. In their experiments, the size of the neighborhood is fixed at 10 by which positive results have been achieved. We also use this value in our experiments for fairness. The regularizer λ in algorithm *AN* should also be carefully chosen. Although various methods have been proposed to deal with this issue, there is currently no reliable way to get the optimal value. Cross validation can be adopted for better performance, however, that would be time-consuming. Therefore we simply fix λ at 1.0 in all our experiments. Actually it will be shown in our experiments that a small change in λ does not affect the performance much.

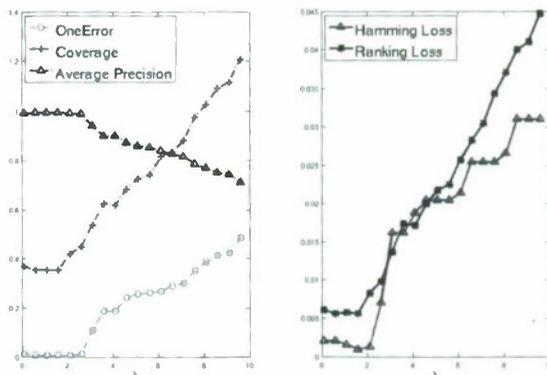


Fig. 2. Indexes' values of *AN* vs. the λ on *genbase*: small λ tends to give better performance which decreases as λ increases, however, a small change at its manually-chosen value(e.g. 1.0 here) does not affect the efficacy much

4.4 Experimental Results and Analysis

First, we test the stability of our *AN* algorithm to the parameter setting of λ by assigning different values of λ in a relative large range on the data sets of *genbase*, as shown in Figure 2. We can see that a small value of λ tends to give better performance. This can be explained that, as λ increases, the optimization will exert more penalty on the sparsity of the w . A very large λ would typically result in very few number(e.g. only 1) of neighbors which are chosen for further classification, which yields a bad classification results. However, it can also be recognized that, for small value of λ , its

Table 2. Comparative Results on *genbase*: *AN* achieves the best performance on all statistic except for Ranking Loss. *IBLR-ML* gets better performance in all statistic than *ML-KNN*.

ALGORITHM	AN	ML-KNN	IBLR-ML
HLOSS ↓	0.0020	0.0050	0.0020
ONEERROR ↓	0.0056	0.0090	0.0070
COVERAGE ↓	0.3518	0.5610	0.4220
RLOSS ↓	0.0058	0.0060	0.0040
AVEPREC ↑	0.9920	0.9890	0.9900

Table 3. Comparative Results on *medical*: *AN* has the best result but for coverage on which *ML-KNN* gets the best performance. On this data set, *IBLR-ML* was surpassed by *ML-KNN* in all statistics.

ALGORITHM	AN	ML-KNN	IBLR-ML
HLOSS ↓	0.0165	0.0171	0.0223
ONEERROR ↓	0.1381	0.2643	0.3844
COVERAGE ↓	1.7177	0.7237	4.7960
RLOSS ↓	0.0253	0.0425	0.0833
AVEPREC ↑	0.8876	0.7957	0.7045

Table 4. Comparative Results on *enron*: Except for Hamming Loss, *AN* achieves the best performance. *ML-KNN* outperforms *IBLR-ML* consistently in all statistics.

ALGORITHM	AN	ML-KNN	IBLR-ML
HLOSS ↓	0.0540	0.0520	0.0572
ONEERROR ↓	0.3005	0.3040	0.3834
COVERAGE ↓	12.8532	13.2055	14.9551
RLOSS ↓	0.0891	0.0938	0.1124
AVEPREC ↑	0.6598	0.6232	0.6020

small change does not affect the performance much. Secondly, we compare our *AN* algorithm with the *ML-KNN* and *IBLR-ML* on the aforementioned measurements. The ↓ beside each measurement means that smaller value yields better performance while ↑ represents the opposite. Table 2 shows the testing results on *genbase*, from which we can see that *AN* algorithm dramatically outperforms the other methods in all statistic except for *Ranking Loss*, on which *IBLR-ML* achieves the best result.

Similarly, Table 3 to Table 5 give the effectiveness of the three algorithms on data sets *medical*, *enron*, *bibtex* respectively. From the experimental results we can see that *IBLR-ML* outperforms *ML-KNN* in data set *genbase* while the opposite results are achieved in data sets *medical* and *enron* and none is guaranteed better than the other. However, although *AN* does not possess the best results in all statistics, it still can be recognized that *AN* dominates the experimental results and outperforms the other two.

Table 5. Comparative Results on *ibtex*: AN leads in all statistics and significantly improvement is achieved in Ranking Loss and Coverage

ALGORITHM	AN	ML-KNN	IBLR-ML
HLOSS ↓	0.0137	0.0140	0.0189
ONEERROR ↓	0.4064	0.5853	0.6294
COVERAGE ↓	26.6282	56.2179	48.7797
RLOSS ↓	0.0896	0.2173	0.1961
AVEPREC ↑	0.5378	0.3449	0.3349

5 Conclusion and Future Work

In this paper, we propose an Adaptive Neighborhood algorithm for multilabel classification. We construct an adaptive neighborhood by an optimization procedure similar to sparse representation but with more interpretability of relation between neighborhood. Based on this automatically-formed neighborhood, classification can be easily carried out. Experiments show our algorithm outperforms the state-of-the-art.

Some issues of this framework should still be ameliorated in the following points which will be our future work:

- The quadratic programming behind the algorithm is time consuming. Solving the optimization more efficiently can be helpful.
- How to take the labels' correlations into account explicitly under the AN framework is another issue.
- Exploring other ways to classification under AN other than our current weighted sum method is desirable.

Acknowledgments

This work was supported by the National Science Foundation of China under the Grant No.60973097 and Higher Education Doctoral Foundation under the grant No.200802870003 respectively. We thank Zhang and Cheng very much for providing the codes of *ML-KNN* and *IBLR-ML*.

References

1. Zhang, M., Zhou, Z.: Ml-knn: A lazy learning approach to multilabel learning. *Pattern Recognition* 40(7), 2038–2048 (2007)
2. Cheng, W., Hullermeier, E.: Combining instance-based learning and logistic regression for multilabel classification. *Machine Learning* 76(2/3), 211–225 (2009)
3. Schapire, R., Singer, Y.: Boostexter: A boosting-based system for text categorization. *Machine Learning* 39(2/3), 135–168 (2000)
4. Elisseeff, A., Weston, J.: A kernel method for multi-labelled classification. *Advances in Neural Information Processing Systems* 14, 681–687 (2002)
5. Wright, J., Yang, A.Y., Ganesh, A., Sastry, S.S., Ma, Y.: Robust face recognition via sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31(2-3), 210–227 (2009)

6. Qiao, L., Chen, S., Tan, X.: Sparsity preserving projections with applications to face recognition. *Pattern Recognition* 43(1), 331–341 (2010)
7. Yang, J., Wright, J., Huang, T., Ma, Y.: Image super-resolution as sparse representation of raw image patches. In: *Computer Vision and Pattern Recognition, CVPR 2008*, pp. 1–8 (June 2008)
8. Elad, M., Aharon, M.: Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image Process.* 15(12), 3736–3745 (2006)
9. Bruckstein, A.M., Donoho, D.L., Elad, M.: From sparse solutions of systems of equations to sparse modeling of signals and images. *SIAM Review* 51(1), 34–81 (2009)
10. Natarajan, B.K.: Sparse approximation solutions to linear systems. *SIAM J. Comput.* 24(2), 227–234 (1995)
11. Chen, S.S., Donoho, D.L., Saunders, M.A.: Atomic decomposition by basis pursuit. *SIAM Review* 43(1), 129–159 (2001)
12. Roweis, S., Saul, L.: Nonlinear dimensionality reduction by locally linear embedding. *Science* 290(22), 2323–2326 (2000)
13. Tenenbaum, J., Silva, V., Langford, J.: A global geometric framework for nonlinear dimensionality reduction. *Science* 290(22), 2319–2322 (2000)
14. Liu, J., Ji, S., Ye, J.: SLEP: Sparse Learning with Efficient Projections. Arizona State University (2009)
15. Diplaris, S., Tsoumakas, G., Mitkas, P., Vlahavas, I.: Protein classification with multiple algorithms. In: Bozanis, P., Houstis, E.N. (eds.) *PCI 2005*. LNCS, vol. 3746, pp. 448–456. Springer, Heidelberg (2005)
16. Katakis, I., Tsoumakas, G., Vlahavas, I.: Multilabel text classification for automated tag suggestion. In: *Proceedings of the ECML/PKDD 2008 Discovery Challenge* (2008)

Time-Sensitive Feature Mining for Temporal Sequence Classification

Yong Yang, Longbing Cao, and Li Liu

Data Sciences & Knowledge Discovery Lab,
Center for Quantum Computation and Intelligent Systems,
Faculty of Engineering & Information Technology,
University of Technology, Sydney
{yongyang, lbcao, liliu}@it.uts.edu.au

Abstract. Behavior analysis received much attention in recent year, such as customer-relationship management, social security surveillance and e-business. Discovering high impact-driven behavior patterns is important for detecting and preventing their occurrences and reducing resulting risks and losses to our society. In data mining community, researchers pay little attention to time-stamps in temporal behavior sequences (without explicitly considering inherent temporal information) during classification. In this paper, we propose a novel Temporal Feature Extraction Method - TFEM. It extracts sequential pattern features where each transition is annotated with a typical transition time (its duration or interval). Therefore it substantially enriches temporal characteristics derived from temporal sequences, yielding improvements in performances, as demonstrated by a set of experiments performed on synthetic and real-world datasets. In addition, TFEM has the merit of simplicity in implementation and its pattern-based architecture can generate human-readable results and supply clear interpretability to users. Meanwhile, it is adjustable and adaptive to user's different configurations, allowing a tradeoff between classification accuracy and time cost.

1 Introduction

Behavior analysis [1,2] is increasingly regarded as a key component in business problem-solving. Unlike traditional analytical methods, behavior informatics is aimed at discovering high impact events (i.e. those activities associated with or causing a specific impact of interest to the business world) from behavioral data. Discovering high impact-driven behavior patterns is important for detecting and preventing their occurrences and reducing resulting risks and losses to our society, such as earthquake prediction, epidemic outbreak monitoring, market surveillance, fraud detection and national security. In order to identify high impact behavior patterns, the usual transactional data needs to be converted into behavioral data, which is organized to explicitly present properties associated with behavior and its impact on business.

A typical situation of recording behavior is through constructing sequences of behavior, and generating so-called sequential data. Sequential data is widely seen in many applications, including business applications and scientific applications. In general, sequential data only involves the ordering relationship existing in behavior sequences.

Table 1. An example dataset of sequences with timestamps

ID	t_1	t_2	t_3	t_4	t_5	t_6	...	label
s_1	a	c	(bd)	c	b	(ac)	...	c_1
s_2	b	a	a	a	a	b	...	c_2
s_3	c	a	a	a	a	(ac)	...	c_2
s_4	a	a	c	c	b	c	...	c_1
s_5	(abc)	a	b	d	e	d	...	c_1

A sequence s_i collects a list of ordered objects e_n , $s_i = \{e_1, e_2, \dots, e_n\}$, in which $e_n = (x_1 x_2 \dots x_q)$ is an element consisting of activities, events or actions in the behavior sequence, and x_q records the properties or items associated with the sequence itemset. When timestamps (t_1, \dots, t_n) are added to their corresponding behavior actions (e_1, e_2, \dots, e_n) , we generate temporal sequences. A temporal sequence is expressed as $s_i = \{(t_1, e_1), (t_2, e_2), \dots, (t_n, e_n)\}$ where $t_{(n-1)} < t_n$. In the real world, a sequence of behavior often incurs certain impact on business, for instance, a series of abnormal online payments incur online payment fraud, a list of high risk terrorist activities may lead to an eventual disaster to the society. Let $C = c_1, c_2, \dots, c_m$ represent such business impacts, c_m is a specific class of impact, for instance, high risk customers. Table 1 shows an example of five sequences, each sequence consists of a list of actions happening at different time points. At some time point, multiple actions co-occur, such as $(t_1, s_5) = (abc)$. Each sequence is associated with a business impact label, for instance, s_5 has associated label c_1 . In practice, quantitative temporal information associated with activities is helpful for distinguishing high impact behavior from others. We call such activities *time sensitive*. Time-sensitive behavior is widely seen in many applications. For instance,

- Example 1. In a medical diagnosis and symptom analysis, the temporal information is crucial for doctors to accurately diagnose diseases. For instance, H1N1 influenza (Swine flu) has a rapid onset within 3-6 hours, presenting with high fever (greater than 102 °F). In contrast, such sudden fever is rare with a common cold. This example shows the importance of considering temporal intervals in sequence analysis.
- Example 2. As for failure detection and identification in assembly line systems, anomaly can be detected with the help of the quantitative temporal intervals between tasks. For example, suppose there are three successive workflow tasks. It is 8 minutes from task 1 to task 2, and 2 minutes from task 2 to task 3. If a record shows 2 minutes from task 1 to task 2 and 6 minutes from task 2 to task 3, apparently this may indicate the presence of anomaly even though the sequence representation of those tasks present nothing abnormal. This example shows that sequence analysis without considering temporal intervals may miss important findings.
- Example 3. In the web usage analysis, if many users tend to stay for a longer time with some particular websites than visiting others, the browsing duration difference indicates more attractive value of the long-stay websites. This example shows the importance of considering user navigation duration in web usage analysis.

To analyze patterns in the above dataset in Table 1 and applications, traditional sequence analysis methods only count the ordering information among sequential items, and treat all actions equally by merging them together. For instance, a health insurer claims one to multiple service types at the same time with increasing frequencies may indicate either increasingly terrible health situation or fraudulent claims. Health insurance providers may be interested in claim review and active customer care, so as to work out why multiple services were conducted at the same time, whether there is any service of the patient's particular interest, why the patient frequently visited doctors, or whether the patient saw different doctors. While these questions are so critical for health insurance providers, it is hard for the existing sequence analysis approaches to find informative hints for these questions.

This is because the existing sequence analysis approaches mainly focus on sequence items, ordering relationship. Consequently, important information in temporal sequence is missing, for instance, the time interval between two consecutive activities, those co-occurring activities at the same time, and the impact label associated with a sequence. However, these aspects are critical for us to disclose in-depth causes and effects associated with discriminative behavior. For this, both temporal sequence analysis and temporal sequence classification can play an important role. Temporal sequence analysis is an emerging research issue in sequence analysis. Limited research has been conducted on mining sequential patterns from temporal sequential data. To the best of our knowledge, current approaches mainly pay attention to the timestamps associated with events, which are converted into sequential orders of the underlying activities.

In addition, while sequence classification is attracting more and more interest [3], people focus on the combination of classification with traditional sequential pattern mining. The goal of sequence classification is to predict which class a given sequence belonged to. No substantial work has been found on classifying temporal sequences.

Unfortunately, how to handle time sensitivity in the temporal sequence classification is a difficult problem. The construction of the sequence of items should be intertwined with the construction of its timestamps. Historically, researchers independently focus on either sequential or temporal aspects. How to combine the temporal information with sequence classification to attain an enhanced informative model is nearly unexploited. In addition, it is very time consuming to identify patterns combining temporal information with sequence classification.

In this paper, we discuss temporal sequence classification. The main idea is to incorporate temporal information into sequence classification. For this, we propose Temporal Feature Extraction Method (TFEM) to mine temporal features for sequence classification. Our contribution is two-fold.

- One is that we design innovative feature mining algorithms which can effectively represent temporal information for sequences classification. The time-sensitive features enrich temporal characteristics derived from the raw data, yielding improvement on sequence classification performance, as demonstrated by a set of experiments performed on synthetic and real-world datasets.
- The other is, our result is easily interpretable. We employ decision tree to generate human-friendly rules. Additionally, it provides an adaptive solution allowing user to determine a tradeoff between classification accuracy and computational cost.

The rest of the paper is organized as follows. Section 2 summarizes related work. Section 3 introduces our novel TFEM approach of mining time-sensitive features for sequence classification. Section 4 presents two empirical studies in which we applied our method to synthetic and real-world datasets. Section 5 discusses an extension of our TFEM approach. Finally we conclude our work in section 6.

2 Related Work

Temporal sequence mining has been explored intensively. Based on the nature of items, sequences can be divided into two categories: symbolic representation (discrete variable e.g. an action code, or tick-by-tick data) and time-series representation (continuous variable e.g. price in the stock market). Here we focus on symbolics as there are multiple approaches to covert time-series data into symbolics: for instance, Discrete Fourier transform (DFT) [4], Singular Value Decomposition (SVD) [5], Adaptive Piecewise constant approximation [6], Symbolic Aggregate Approximation (SAX) [7].

There are enormous renowned classification algorithms. However, they are difficult to apply to sequential data, because there could be huge features potentially and thus intractable for relatively limited computing resources. In a seminal paper, Lesh etc. [8] proposed *FeatureMine* for sequence classification by analysing the presence of features derived from discriminative frequent patterns. The three phases of Lesh's method are:

1. Mining features. First of all, it adapts SPADE [9] to generate frequent patterns from sequence data. Chi-square tests are used to prune patterns to enforce discriminative and redundancy constraints. Remaining patterns f_1, f_2, \dots, f_n are outputted as features for classification.
2. Applying features to sequences. Most standard classifiers only accept an example as input when it is in the form of a vector consisting of feature-value pairs. Each feature generates a boolean value depending on its presence in a sequence. For example, if sequence s_i is "in presence of" pattern f_1 (i.e., f_1 is a subsequence of s_i), the value with regard to feature f_1 is true, otherwise it is false.
3. Classification. Based on the boolean feature-value pairs, traditional attribute-based classifiers can be used, such as Winnow and Naive Bayes.

After that, [10,11,12] incorporate biological knowledge into DNA sequence classification. Recently, there are overwhelming tools on protein sequences [13,14,15]. [16] uses implicit motif distribution based hybrid computational kernel for sequence classification. But to our best knowledge, combining sequence classification with temporal information is nearly unexploited.

3 A Novel TFEM Approach

As discussed in previous section, most existing sequence classification approaches seldom explicitly take time intervals between items into consideration. To address this limitation, we propose temporal feature extraction method (TFEM) to capture the interval characteristic.

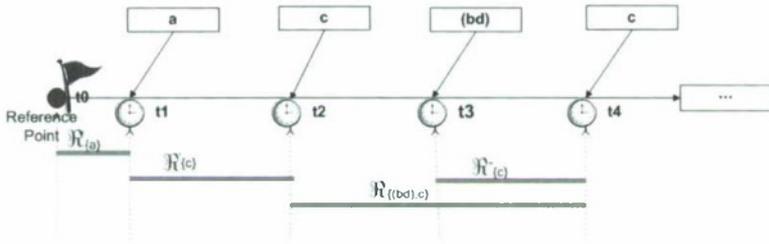


Fig. 1. A timeline representation of partial sequence s_1 from t_1 to t_4

Definition 1. A behavior sequence $s_i = \{e_1, e_2, \dots, e_n\}$, in which $e_n = (x_1 x_2 \dots x_q)$ is an atomic item consisting of activities, events or actions in the behavior sequence, and x_q records the properties or items associated with the sequence itemset. If $q = 1$, e_n is a single atomic item, otherwise it is composite atomic item.

Definition 2. For an atomic item e_n , $t_c[e_n]$, $t_a[e_n]$ denote the time stamps of current item and previous item a sequence s_i , respectively. In particular, for the first item in s_i , $t_a = t_0$, which is a **reference time** or start point for calculation.

Definition 3. If a pattern p contains only one atomic item, p is called 1-itemset; otherwise we name the first item in p as $p_{firstItem}$ and the last item as $p_{lastItem}$. An interval \mathcal{R} for pattern p in sequence s_i is defined as

$$\mathcal{R} = \begin{cases} Avg(t_c[p] - t_a[p]), & p \text{ is 1-itemset,} \\ Avg(t_c[p_{lastItem}] - t_a[p_{firstItem}]), & \text{Otherwise.} \end{cases} \quad (1)$$

$$\mathcal{R}'_{\{c\}} = \begin{cases} Avg(t_c[p] - t_a[p]), & p \text{ is 1-itemset,} \\ Avg(t_c[p_{lastItem}] - t_a[p_{firstItem}]), & \text{Otherwise.} \end{cases} \quad (1')$$

If pattern p repeats in s_1 , an average value is taken when calculating \mathcal{R} .

An example of calculating intervals within s_1 from t_1 to t_4 is depicted in Fig. 1. For instance, for 1-itemset $\{a\}$, $\mathcal{R}_{\{a\}} = t_1 - t_0$. For 2-itemset $\{(bd), c\}$, $p_{firstItem}$ is $\{(bd)\}$ and $p_{lastItem}$ is $\{c\}$. Therefore, $v_{\{(bd), c\}} = t_4 - t_2$. Again for 1-itemset pattern $\{c\}$, it occurs twice in s_1 . For the first presence of $\{c\}$, the interval $\mathcal{R}'_{\{c\}} = t_2 - t_1$ and for the second presence, $\mathcal{R}''_{\{c\}} = t_4 - t_3$. $\mathcal{R}_{\{c\}} = (\mathcal{R}'_{\{c\}} + \mathcal{R}''_{\{c\}})/2 = (t_4 - t_3 + t_2 - t_1)/2$.

The basic idea of our TFEM approach is during the traditional feature extraction for sequence classification, we incorporate interval information to create more informative features and thus classifier can take advantage of those constructed new TFEM features.

3.1 Framework

The dataflow of our TFEM sequence classification is described in Figure 2. The whole process is divided into three phrases:

- **Data Representation and Preprocessing:** First of all, sequential pattern mining algorithm is employed to get initial features (Basically they are frequent patterns extracted from raw data and have been pruned by statistical tests). Then we calculate an interval for each pattern in each sequence. Thus we can generate 2-tuple (pattern, interval) pairs for every sequence.

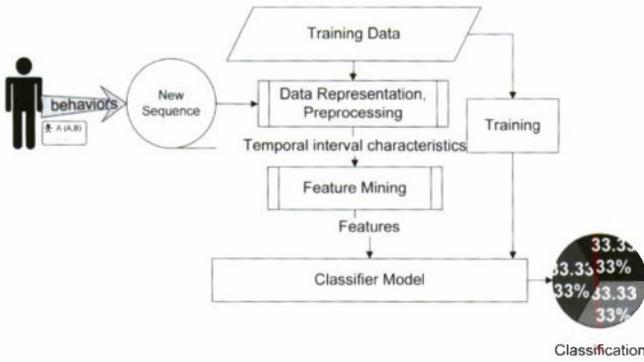


Fig. 2. A dataflow of behavior sequence classification

- **Feature Mining:** The TFEM algorithm in section 3.3 is designed to construct new temporal features for sequence classification.
- **Training and Testing:** 10 fold cross-validation is conducted. Decision tree classifier is used to generate easily interpretable rules. Then the trained classifier makes predictions on incoming sequences.

3.2 Data Representation and Preprocessing

By using featureMine proposed by Lesh etc. [8], we attain patterns $\{a\}$, $\{(ac)\}$, $\{b\}$, $\{a, b\}$, ... as our initial features in the previous example. Then for each pattern f_i in every sequence we calculate its interval using formula 1 and generate 2-tuple (pattern, interval) pair, which is shown in table 2.

Table 2. An example dataset in (pattern, interval) pairs

ID	$\{a\}$	$\{b\}$	$\{a, b\}$	$\{(ac)\}$...
s_1	1	1	2	1	...
s_2	1	1	3		...
s_3	1	1	3	3	...
s_4	1	1	2		...
s_5	1	1	2	1	...

3.3 Feature Mining

Construction of Temporal Features. We design TFEM temporal feature algorithm to construct new temporal features, which is described in Fig. 3.

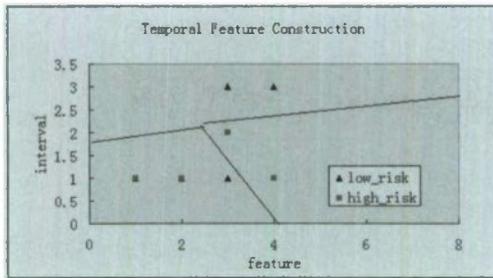


Fig. 3. A example of temporal feature construction

Algorithm 1: Temporal Feature Extraction Algorithm

Input: $\text{Min_freq}(c_i)$, Dataset D.

Output: Candidate temporal features.

- (a). Represent data as 2-tuple (pattern, interval) pairs in the two dimensional feature space.
- (b). Merge and eluster. In order to reduce the feature space, for the same pattern p , intervals are merged if they belong to the same class. For example, if any feature examples generated in previous step from (pattern 1, 8) to (pattern 1, 10) are all positive, they can be merged as (pattern 1, 8~10). For those belonging to multiple classes, we adopt an odds-ratio test and simply prune points which are less skewed in the class distribution. For example, in two-classes classification, we calculate the discriminative power by the following formula:

$$E = \frac{p_1/(1-p_1)}{p_2/(1-p_2)} \quad (2)$$

where p_1, p_2 are proportions of a pattern in difference classes respectively. Divide our (pattern, interval) space into several regions by clustering. It is shown that there are three regions in Fig. 3.

- (c). Output region boundaries as candidate temporal features. In our example, the three regions are our newly constructed temporal features.
-

The next step is to make use of these regions. For an incoming sequence, we check every pattern's presence. If the pattern occurs then calculate its interval value and locate its point in (pattern, interval) two-dimension feature space. The temporal feature value is true or false depending on which region it falls in.

Temporal Feature Selection. After constructing new temporal features, statistical optimization is performed in order to achieve highly efficient classification. There are three pruning criteria in our algorithm:

1. Features should be frequent and with strong discriminative power.
2. Features should be efficient for classification.
3. Features should be optimized, without complex parameter tuning.

This process is described in algorithm 2.

Algorithm 2: Temporal Feature Mining Algorithm

Input: Dataset D in the form of (pattern, interval) pair.

Output: Temporal features.

- (a). Generate candidate features by previous feature extraction algorithm.
- (b). Prune any candidate if it meets any criterion in the following tests:
 - Discriminative test: The odds-ratio test is employed to ensure features are significantly discriminative among classes.
 - Redundancy test: We create new calculation formula based on Foil-Gain [17] to estimate information gain. For instance, regarding to biclassification

$$E = \text{Max}(tw(\log_2 \frac{p_1}{p_1 + n_1} - \log_2 \frac{p_2}{p_2 + n_2})) \quad (3)$$

where p_1, n_1 is that number of positive and negative examples covered before adding new feature. p_2, n_2 is that number of positive and negative examples covered when adding one new feature. t is the number of positive examples covered by both. w is the proportion of pattern's duration time in global temporal dimension.

- Optimization test: We tune our model by enumerate parameters' thresholds. For instance, the threshold for pattern's length can be determined by simply the trial and error method, that is, running our tests with different length and selecting the best.
- (c). Output newly constructed features after pruning in step b.
-
-

3.4 Training and Testing

We choose a rule-based classification method for several reasons. First, it generates human-readable results. This is very important for the interpretability of our model in practice. Secondly, it is efficient. The time complexity is $O(N)$ while N is the number of rules. Finally, with respect to imbalance data, rule-based learner is more effective.

Based on our temporal features, classifier can improve its accuracy as those constructed features help to capture informative temporal characteristics in the raw data.

4 Empirical Studies

In order to evaluate our methods, we implement TFEM in both symbolic sequences and time-series datasets.

4.1 Health Insurance Dataset

We use a health insurance dataset to test our TFEM framework, which describes every member's (or user's) claim history. In our experiment, there are a total of 15875 records from 479 users. Each record is in the format of 4-tuple vector (member_id, service_date, service_code, server_content). We reorganize the data into sequences based on the attribute of *member_id* in a temporal order. This dataset contains a sample of

Table 3. Traditional sequence classification confusion matrix

accuracy: 76.41 %			
	true high-risk	true low-risk	class precision
pred. high-risk	198	71	73.61%
pred. low-risk	42	168	80.00 %
class recall	82.50 %	70.29%	

Table 4. TFEM sequence classification confusion matrix

accuracy: 83.11 %			
	true high-risk	true low-risk	class precision
pred. high-risk	183	24	88.41 %
pred. low-risk	57	215	79.04%
class recall	76.25 %	89.96%	

479 sequences with unequal length. Each sequence depicts a member's claim history. Besides, each sequence in the training set has been labeled as either "high-risk" or "low-risk". Table 1 shows a sample of our dataset. For privacy preserving, a, b, c denote the abstraction of actions in each real-world sequence record. c_1 represents high-risk class label while c_2 is low-risk class label. Apparently, two items may happen in the same time. For example, in sequence s_1 , a and e are both associated with time-stamp t_6 .

Our algorithms are developed by Java 1.6, under Eclipse 3.2 environments. Hardware of our computer is duo-core Intel Pentium 4.2 with 1.5 G memory.

We conduct sequence classification on the insurance data. After frequency pattern mining phrase, we obtain 80 features with $\text{min_support}=48$. The art for choosing an appropriate min_support threshold is to make sure our feature set is neither too big nor too small. In this discriminative test, the parameter value of odd-rate is 2. We use 10-fold cross validation and calculate classification accuracy. Table 3 describes the performance of Lesh's method as a benchmark. By comparison, table 4 shows the performance of TFEM model. From the performance contrast test, we can see the TFEM framework can increase the accuracy from 76.41% to 83.11%.

4.2 Ionosphere Dataset

The ionosphere dataset is downloaded from UCI KDD repository [18]. The time-series data was collected by a system in Goose bay, Labrador. There are two classes in a total of 351 samples. After converting those time series data, we run traditional frequent pattern based sequence classification and our TFEM approach. The result shows TFEM outperforms its conventional counterpart with an increase in accuracy from 76.13% to 81.09%.

4.3 Effects of Varying Odds-Ratio

Fig. 4 shows comparison of traditional method and TFEM under several odds-ratio parameter settings. We adjust different odds-ratio and measure the accuracy and time-cost.

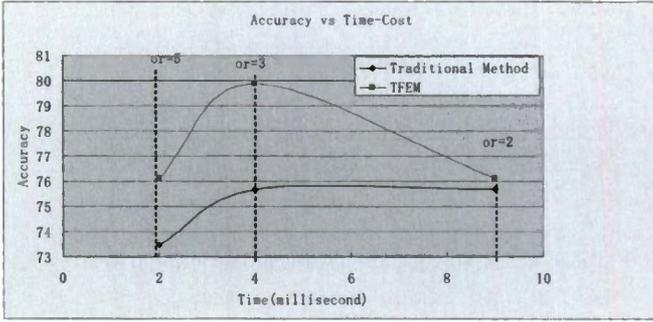


Fig. 4. Accuracy vs. time-cost

It is observed that the greater the value of odds-ratio parameter is, the more candidate features pruned, which reduces overall time-cost. On the other hand, higher accuracy will lead to longer feature extraction time. Flexibility is offered with a tradeoff between classification accuracy and time-cost.

5 Discussion

In this section, we first employ PCA [19] to reduce the computation cost in our algorithms and make TFEM more efficient. Then we discuss about handling time-series data.

Principal component analysis (PCA) describes a mathematical procedure that transforms a number of possibly correlated variables into a smaller number of uncorrelated variables called principal component. PCA was first invented in 1901 by Karl Pearson [20]. PCA [21,22] is mathematically an orthogonal linear transformation that transforms data to a new coordinate system. As you can see from our insurance experiment, there are 23 features. In some cases, in order to find better fine granularity for frequent patterns, we may end up with hundreds of features. Therefore, PCA is used to optimize our model. Fig. 5 depicts the cumulative proportion of variance. In this way, the number

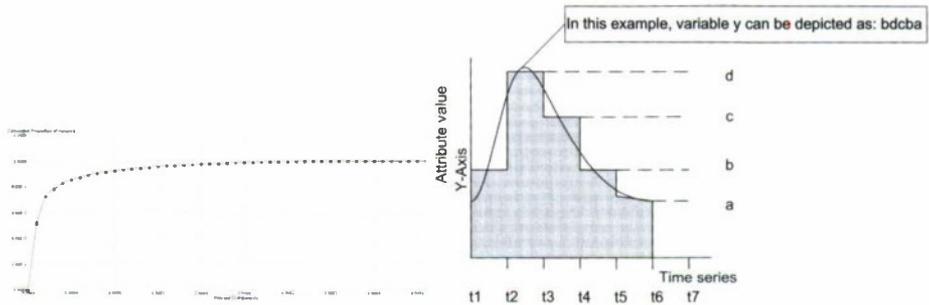


Fig. 5. Principal components analysis and shift to symbolic events

of features can be significantly reduced and only the most representative instances are kept.

Fig. 5 also shows how to convert continuous variables into the symbolic representation. This method is based on Yi and Faloutsos and Keogh et al.'s Piecewise Aggregate Approximation (PAA) [23]. In PAA, each record of time series data is divided into k segments with equal length and the average value of each segment is used as data-reduced representation. Obviously the PAA model is very straightforward and easy to implement. It is very fast and has almost linear time complexity. But on the other hand, it may lose useful information and a variable indicating the slope in each segment becomes useful during the conversion process.

6 Conclusion

Quantitative temporal information associated with activities is helpful for distinguishing high impact behavior from others in many business problem-solving. In this paper, we proposed a novel temporal feature extraction for behavior sequence classification. TFEM incorporates time intervals, which are critical in many business applications, into behavior sequence classification. With informative features, experiments show the performance of classifier is significantly improved.

TFEM is of great significance for discovering knowledge from time-sensitive behavior sequences. Furthermore, it is important to note that TFEM can be easily extended to handle other characteristics without being limited to temporal dimension, such as spatial space.

Acknowledgments. This work is sponsored in part by Australian Research Council Discovery Grants (DP1096218, DP0988016, DP0773412) and ARC Linkage Grant (LP0989721, LP0775041).

References

1. Foxall, C., James, V.: Behavior Analysis of Consumer Brand Choice: A Preliminary Analysis. *The Behavioral Economics of Brand Choice*, p. 54 (2007)
2. Cao, L.: Behavior informatics and analytics: Let behavior talk. In: *ICDM Workshops*, pp. 87–96. IEEE Computer Society, Los Alamitos (2008)
3. Lesh, N., Zaki, M.J., Ogihara, M.: Mining features for sequence classification. In: *KDD 1999: Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 342–346. ACM, New York (1999)
4. Brigham, E., Yuen, C.: The fast Fourier transform. *IEEE Transactions on Systems, Man and Cybernetics* 8(2), 146–146 (1978)
5. Golub, G., Reinsch, C.: Singular value decomposition and least squares solutions. *Numerische Mathematik* 14(5), 403–420 (1970)
6. Eriksson, K., Estep, D., Hansbo, P., Johnson, C.: Introduction to adaptive methods for differential equations. *Acta numerica* 4, 105–158 (2008)
7. Lin, J., Keogh, E., Wei, L., Lonardi, S.: Experiencing SAX: a novel symbolic representation of time series. *Data Mining and Knowledge Discovery* 15(2), 107–144 (2007)

8. Lesh, N., Zaki, M., Ogihara, M.: Mining features for sequence classification. In: Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 342–346. ACM, New York (1999)
9. Zaki, M.: SPADE: An efficient algorithm for mining frequent sequences. *Machine Learning* 42(1), 31–60 (2001)
10. Ma, Q., Wang, J., Shasha, D., Wu, C.: DNA sequence classification via an expectation maximization algorithm and neural networks: a case study. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews* 31(4), 468–475 (2001)
11. Rätsch, G., Sonnenburg, S., Schäfer, C.: Learning interpretable SVMs for biological sequence classification. *BMC bioinformatics* 7(Suppl. 1), S9 (2006)
12. Ferreira, P., Azevedo, P.: Protein sequence classification through relevant sequence mining and bayes classifiers. In: Bento, C., Cardoso, A., Dias, G. (eds.) *EPIA 2005. LNCS (LNAI)*, vol. 3808, pp. 236–247. Springer, Heidelberg (2005)
13. Mulder, N., Apweiler, R.: InterPro and InterProScan: tools for protein sequence classification and comparison. *Methods in Molecular Biology (Clifton, NJ)* 396, 59 (2007)
14. Shen, L., Satta, G., Joshi, A.: Guided learning for bidirectional sequence classification. In: *Annual Meeting-Association for Computational Linguistics*, vol. 45, p. 760 (2007)
15. Spurdle, A., Lakhani, S., Healey, S., Parry, S., Da Silva, L., Brinkworth, R., Hopper, J., Brown, M., Babikyan, D., Chenevix-Trench, G., et al.: Clinical classification of BRCA1 and BRCA2 DNA sequence variants: the value of cytokeratin profiles and evolutionary analysis—a report from the kConFab Investigators. *Journal of Clinical Oncology* 26(10), 1657 (2008)
16. Atalay, V., Cetin-Atalay, R.: Implicit motif distribution based hybrid computational kernel for sequence classification. *Bioinformatics* 21(8), 1429–1436 (2005)
17. Quinlan, J.: Learning logical definitions from relations. *Machine learning* 5(3), 239–266 (1990)
18. Uci kdd repository,
<http://archive.ics.uci.edu/ml/datasets/Ionosphere>:
19. Jolliffe, I.: *Principal component analysis*. Springer, Heidelberg (2002)
20. Gorban, A., Kgl, B., Wunsch, D., Zinovyev, A.: *Principal manifolds for data visualization and dimension reduction*, p. 340. Springer Publishing Company, Heidelberg (2007) (incorporated)
21. Rohlf, F.: Morphometric spaces, shape components and the effects of linear transformations. In: *Advances in morphometrics*, pp. 117–129 (1996)
22. Cai, D., He, X., Han, J., Zhang, H.: Orthogonal laplacianfaces for face recognition. *IEEE Transactions on Image Processing* 15(11), 3608–3614 (2006)
23. Keogh, E., Pazzani, M.: Scaling up dynamic time warping for datamining applications. In: *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 285–289. ACM, New York (2000)

Learning Automaton Based On-Line Discovery and Tracking of Spatio-temporal Event Patterns

Anis Yazidi¹, Ole-Christoffer Granmo¹, Min Lin*, Xifeng Wen*,
B. John Oommen^{1,2}, Martin Gerdes³, and Frank Reichert¹

¹ Dept. of ICT, University of Agder, Grimstad, Norway

² School of Computer Science, Carleton University, Ottawa, Canada**

³ Ericsson Research, Aachen, Germany

Abstract. Discovering and tracking of spatio-temporal patterns in noisy sequences of events is a difficult task that has become increasingly pertinent due to recent advances in ubiquitous computing, such as community-based social networking applications. The core activities for applications of this class include the sharing and notification of events, and the importance and usefulness of these functionalities increases as event-sharing expands into larger areas of one's life. Ironically, instead of being helpful, an excessive number of event notifications can quickly render the functionality of event-sharing to be obtrusive. Rather, any notification of events that provides redundant information to the application/user can be seen to be an unnecessary distraction. In this paper, we introduce a new scheme for discovering and tracking noisy spatio-temporal event patterns, with the purpose of suppressing reoccurring patterns, while discerning novel events. Our scheme is based on maintaining a collection of hypotheses, each one conjecturing a specific spatio-temporal event pattern. A dedicated Learning Automaton (LA) – the *Spatio-Temporal Pattern LA* (STPLA) – is associated with each hypothesis. By processing events as they unfold, we attempt to infer the correctness of each hypothesis through a real-time guided random walk. Consequently, the scheme we present is computationally efficient, with a minimal memory footprint. Furthermore, it is ergodic, allowing adaptation. Empirical results involving extensive simulations demonstrate the STPLA's superior convergence and adaptation speed, as well as an ability to operate successfully with noise, including both the erroneous inclusion and omission of events. Additionally, the results included, which involve a so-called "Presence Sharing" application, are both promising and in our opinion, impressive. It is thus our opinion that the proposed STPLA scheme is, in general, ideal for improving the usefulness of event notification and sharing systems, since it is capable of significantly, robustly and adaptively suppressing redundant information.

Keywords: Learning Automata, Spatio-Temporal Pattern Recognition.

* Former student. Can be contacted at: C/o Dr. Ole-Christoffer Granmo, Dept. of ICT, University of Agder, Grooseveien 36, 4876 Grimstad, Norway.

** *Chancellor's Professor; Fellow : IEEE and Fellow : IAPR.* The Author also holds an *Adjunct Professorship* with the Dept. of ICT, University of Agder, Norway.

1 Introduction

Presence Sharing is a ubiquitous service in which distributed mobile devices periodically broadcast their identity via short-range wireless technology such as Bluetooth or WiFi [1]. The whole problem of *Presence Sharing* is intricately bound to the issue of the recording and processing of “events” involving the entities included within the social network. Applications that utilize *Presence Sharing* have been used in social contexts to maintain an “in touch” feeling strengthening social relations [2], as well as in work environments to enhance collaboration between colleagues [3].

Typically, “events” occurring in the real world can be characterized as being in one of two classes, i.e., “Stochastically Episodic” (SE) and “Stochastically Non-Episodic” (SNE). This is a distinction that is especially pertinent in simulation, where it is customary for one to model the behaviour of accidents, telephone calls, network failures etc. using their respective probability distributions, even though they follow no known pattern. Indeed, events of these families happen all the time, and so can be termed as being “stochastically non-episodic”. As opposed to this, there is a whole class of events that can stochastically occur in a non-anticipated manner. These so-called “stochastically episodic” events include earthquakes, nuclear explosions etc. The difficulty with modelling SE events is that most of the observations appear as noise. However, when the SE event does occur, its magnitude and features far overshadow the background, as one observes after a seismic event. The modelling and simulation of such SE events in the presence of a constant stream of SNE events is a relatively new field [4,5], where the authors model the SE and SNE events *simultaneously* in such a way that the effect of an SE event is perceived through the “lens” of the underlying background of SNE events.

Since events are almost omnipresent, one has to consider the observation due to Garlan *et al.* [6], who state that the most precious resource in a computer system is no longer its processor, memory, disk, or network, but rather human attention. Thus, our aim in this paper is to address a fundamental challenge concerning the above class of applications: *How can one harvest the benefit of event-sharing without distracting the application user with redundant notifications?* The solution we propose is to try to discern the nature of the events encountered¹. Of course, the events may not be drastically SE or SNE, as in the case of earthquakes or nuclear explosions. However, if we can discern that an event is repeating (even though this repetition is non-periodic), it is still of a SNE nature which must be given less weight, while non-repeating events (which are in one sense, SE) must be assigned a greater weight. Thus, the question we resolve involves demonstrating how we can enhance the *Presence Sharing* experience by weighting the SE and SNE events appropriately.

¹ To exemplify the usefulness of such a strategy, consider the nuisance caused by being notified every time one meets a colleague at work, which is a repeating pattern, or a SNE event. In contrast, it would be far more useful to be promptly notified whenever the same colleague unexpectedly appears in your vicinity after a travel abroad. This would be non-repeating pattern, or an SE event.

1.1 Related Work

A number of earlier studies have investigated techniques for discovering the periodicity of time patterns, such as the episode² discovery algorithm found in [7]. However, episode discovery, and other related approaches, suffer from the limitation that they assume unperturbed patterns that exhibit an exact periodicity. Unfortunately, the real-life unfolding of events is typically noise ridden. On the one hand, regular events may get cancelled, introducing what we define as *omission noise*, and on the other, events may arise spontaneously and unexpectedly, without being part of a periodic pattern, introducing *inclusion noise*. A pioneering work which was reported in [8], introduced the concept of *off-line* mining of partially periodic events. Nevertheless, deciding whether to suppress event notifications must often be done instantaneously, as the events are unfolding. Indeed, we argue that any realistic scheme should discover and adapt to patterns as they appear and evolve in an on-line manner, without relying on extensive off-line data mining.

1.2 Paper Contribution and Organization

The paper is organized as follows. In Section 2, we present our overall approach to on-line discovery and tracking of spatio-temporal event patterns, in which the so-called Learning Automata (LA) plays a crucial role. The scheme is designed to deal with noisy spatio-temporal event patterns, when event patterns are evolving with time. We continue in Section 3 by evaluating our scheme using an extensive range of static and dynamic *noisy* event patterns. The experiments demonstrate the scheme's superior convergence and adaptation speed, as well as an excellent ability to operate successfully with noise, including both erroneous inclusion and omission of events. In order to highlight the applicability of our scheme, we present a "*Presence Sharing*" application prototype in Section 4 where we also summarize some initial user experiences. Finally, Section 5, concludes the paper and also provide pointers for further work.

2 On-Line Discovery and Tracking of Spatio-temporal Event Patterns

The method which we propose is based on the theory of LA. Since space does not permit a detailed overview of this theory, this is included elsewhere [9]. However, in all brevity, we state that our scheme is based on maintaining a collection of hypotheses, each one conjecturing a specific spatio-temporal event pattern. A dedicated LA, which we coin the *Spatio-Temporal Pattern LA* (STPLA), is associated with each hypothesis. The STPLA decides whether its corresponding hypothesis is true by observing events as they unfold, processing evidence for and/or against the correctness of the hypothesis. To explain this, we first address hypothesis management, and then proceed with the details of the STPLA.

² The expression "episode" used in this setting must not be confused with the class of SE and SNE events described in the earlier paragraph.

2.1 Hypothesis Management

The premise of our discussions is the following: In order to reduce distraction, events should only be signalled when they are SE. This means that they cannot be anticipated, obey no known stochastic distribution, and possess an element of “surprise”, i.e., they can not be easily predicted by the recipient³. An event can either be sporadic, arising spontaneously, or it can be part of a spatio-temporal pattern, making it occur regularly. In either case, if it cannot be explained by any of the spatio-temporal patterns that are known by the recipient, the recipient should be notified. However, when the event constitutes a part of an ongoing spatio-temporal pattern, it is really non-episodic (or SNE) in nature. We require that this phenomenon be discovered as soon as possible, so that the events generated from this pattern can be suppressed before the pattern loses its novelty to the recipient.

In our proposed scheme, when an event is observed, all potentially interesting patterns that *could* have produced the event are identified. We refer to these potential patterns as *hypotheses*. The reader will thus observe that our approach is based on the concept of predefined pattern structures, as advocated in [10], rather than trying to look for patterns with unknown structure. Thus, in this spirit, we consider a discrete world of m spatial location primitives $L = \{l_1, l_2, \dots, l_m\}$ and of n discrete time primitives $T = \{t_1, t_2, \dots, t_n\}$ of appropriate granularity. By way of example, the location primitives could be “Home”, “Office”, or “Abroad”, while the time primitives could be “Mondays”, “Tuesdays”, “Weekends”, and so on. The location and time primitives are combined from their cross-product spaces to produce spatio-temporal patterns. Thus, the resulting spatio-temporal pattern space would (or could) be an exhaustive enumeration of relevant combinations such as “Mondays at Office”, “Weekends at Office”, and so on. Each spatio-temporal pattern of the latter form is seen as a hypothesis, conjecturing that the respective pattern specifies an ongoing stream of events. In the following, we assume that there are r such hypotheses, represented as a set $H = \{h_1, h_2, \dots, h_r\}$. Observe that although the cardinality of this set might get large, the computational efficiency and small memory footprint of our LA (as seen presently), effectively handles the size of the state space.

Note too that the novelty of this present work is not the above indicated structuring of the spatio-temporal pattern space, which is a well-known approach used in typical calendar systems. Rather, it is the learning scheme we propose⁴ for determining whether a given spatio-temporal event pattern can be found in a stream of events, in an on-line manner, and under noisy conditions.

³ Events should, of course, also match the interest profile of the recipient. We will, in this paper, assume that all events are of interest, as long as they are novel. On-line adaptive learning of interest profiles will be addressed in another forthcoming paper.

⁴ Using the techniques presented in [4,5], we are currently investigating how one-class classifiers can be used to learn the most appropriate hypothesis. This would assume that the patterns which can be anticipated constitute the SNE events, and the set of SE events, which cannot be anticipated, constitutes the one-class to be recognized.

2.2 Learning Automaton Based On-Line Discovery and Tracking of Spatio-temporal Event Patterns

We base our work on the principles of LA [9,11]. LA have been used to model biological systems [12], and have recently attracted considerable interest because they can learn the optimal actions when operating in (or interacting with) unknown stochastic environments. Furthermore, they combine rapid and accurate convergence with low computational complexity.

Generally stated, an LA chooses a sequence of actions offered to it by a random *environment*. The environment can be seen as a generic *unknown* medium that responds to each action with some sort of reward or penalty, usually *stochastically*. Based on the responses from the environment, the aim of the LA is to find the action that minimizes the expected number of penalties received. Before we proceed with describing the STPLA itself, it is necessary for us to first define the environment that we are dealing with.

Spatio-Temporal Pattern Environment: The purpose of the Spatio-Temporal Pattern Environment is to provide feedback to the individual STPLA about the validity of their respective hypotheses.

In all brevity, at each time instant matching the time primitive t_i , if an STPLA predicts the presence of an event at location l_j , it informs the environment about this prediction. Conversely, if the STPLA predicts the absence of an event at the same location, this too is submitted to the environment. The environment, in turn, responds with a *Reward* if an event took place (or did not take place) as predicted. If the prediction is incorrect, on the other hand, the environment responds with a *Penalty* instead. That is, the STPLA is penalized if an event takes place, but none was predicted, or if an event is predicted, but does not take place. The latter reward policy is illustrated in Fig. 1.



Fig. 1. Feedback for a daily event hypothesis (R-Reward, P-Penalty)

The figure illustrates events generated from a daily meeting. The STPLA that hypothesizes a daily meeting will be rewarded each day a meeting takes place (green circle) because of its ability to correspondingly predict the daily event. An important challenge that we address in this paper, however, is how to deal with spatio-temporal event patterns that are affected by noise. In the figure, for example, some of the daily meetings may be cancelled (depicted by white circles) due to external conditions, such as when the participants are unavailable. Thus, when meetings are cancelled, the STPLA maintaining the daily meeting hypothesis will get penalized because of its prediction, despite the fact that its hypothesis is true. In a similar vein, so-called "straggler" events, not being part of any periodic pattern, can also occur in a sporadic and spontaneous manner.

From the above example it can be seen that we face two kinds of noise:

Omission Error: This is an error which occurs when an event that forms a part of a periodic spatio-temporal pattern is randomly left out. In other words, the event was supposed to have taken place according to the pattern, but did not. Notice the SE nature of this event – it is not something that could have been anticipated.

Inclusion Error: This is an error which occurs when an event that occurs is not part of a periodic (anticipated) pattern, but rather arises sporadically and spontaneously. Again, one must observe the SE nature of this event.

By way of example, Alice may cancel a regular meeting with Bob due to ill health. However, Alice may still meet Bob sometime outside of the regular meeting schedule – purely by chance (e.g., an accidental meeting in the canteen). In this manner, we can appropriately model both these kinds of noise.

The Spatio-Temporal Pattern Learning Automaton (STPLA): We now introduce the STPLA that we have designed to discover and track spatio-temporal patterns. In brief, the task of an STPLA is to decide whether a specific spatio-temporal pattern hypothesis is true. By observing events as they unfold, the correctness of an hypothesis is decided.

The STPLA can be designed to model arbitrarily general SE and SNE events. But due to space limitations, in this paper, we confine our design and implementation details to events which can be characterized *deterministically*.

The STPLA is inspired by so-called family of fixed structured LA [13]. Accordingly, a STPLA can be defined in terms of a quintuple [9]:

$$\{\underline{\Phi}, \underline{\alpha}, \underline{\beta}, \mathcal{F}(\cdot, \cdot), \mathcal{G}(\cdot, \cdot)\}.$$

Here, $\underline{\Phi} = \{\phi_1, \phi_2, \dots, \phi_s\}$ is the set of internal automaton states. $\underline{\alpha} = \{\alpha_1, \alpha_2, \dots, \alpha_r\}$ is the set of automaton actions. Further, $\underline{\beta} = \{\beta_1, \beta_2, \dots, \beta_m\}$ is the set of inputs that can be given to the automaton. An output function $\alpha_t = \mathcal{G}[\phi_t]$ determines the action performed (or chosen) by the automaton given the current automaton state. Finally, a transition function $\phi_{t+1} = \mathcal{F}[\phi_t, \beta_t]$ determines the new state of the automaton from: (1) The current state of the automaton and (2) The response of the environment to the action performed (or chosen) by it.

Based on the above generic framework, the crucial issue is to design automata that can learn the optimal action when interacting with the environment. Several designs have been proposed in the literature, and the reader is referred to [9] for an extensive treatment. In this paper, since we target the learning of spatio-temporal patterns, our goal is to design an LA that is able to discover and track such patterns over time. Briefly stated, we construct an automaton with

- States: $\underline{\Phi} = \{1, 2, \dots, N_1, N_1 + 1, \dots, N_1 + N_2 + 1\}$.
- Actions: $\underline{\alpha} = \{Notify, Suppress\}$.
- Inputs: $\underline{\beta} = \{Reward, Penalty\}$.

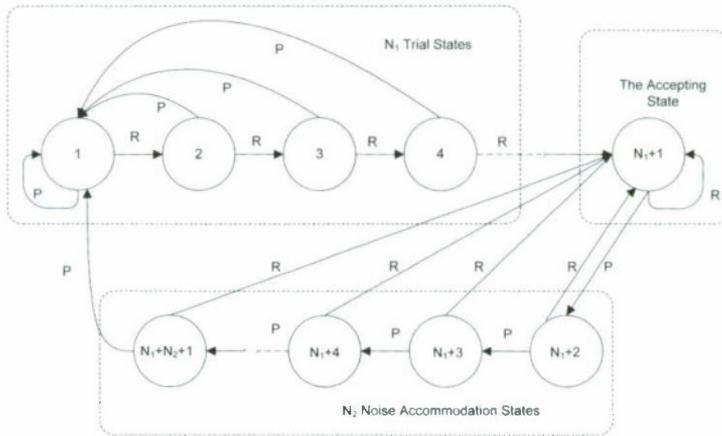


Fig. 2. The state transition map and the output function of a STPLA

Fig. 2 specifies the state space of STPLA as well as the \mathcal{G} and \mathcal{F} matrices. The \mathcal{G} matrix can be summarized as follows. If the automaton state lies in the set $\{1, \dots, N_1\}$, which we refer to as the *Pattern Evaluation States*, then the LA will choose the action "Notify". If, on the other hand, the state is either $N_1 + 1$ or one of the states in the set $\{N_1 + 2, \dots, N_1 + N_2 + 1\}$, it will choose the action "Suppress". We refer to the state $N_1 + 1$ as the *Pattern Acceptance State*, and the states $\{N_1 + 2, \dots, N_1 + N_2 + 1\}$ as the *Pattern Tracking States* for reasons explained presently. Note that since we initially do not know whether a pattern is present, we set the initial state of our automaton to 1.

The state transition matrix \mathcal{F} determines how the learning proceeds. In brief, the learning is divided into three parts:

Pattern Evaluation: In the Pattern Evaluation part, the goal of the LA is to discover the presence of the spatio-temporal event pattern associated with the maintained hypothesis, without being distracted by omission and inclusion errors. In this phase, the state transitions illustrated in the figure are such that any deviance from the hypothesized pattern, modelled as a Penalty (P), causes a jump back to state 1. Conversely, only a systematic presence of the pattern hypothesized, modelled as a pure sequence of Rewards (R), will allow the LA to pass into the Pattern Acceptance part.

Pattern Acceptance: In the Pattern Acceptance part, consisting of state $N_1 + 1$, the hypothesized pattern has been confirmed with high probability.

Pattern Tracking: In the Pattern Tracking part, consisting of states $\{N_1 + 2, \dots, N_1 + N_2 + 1\}$, the goal is to detect when the discovered pattern disappears, without getting distracted by omission errors. Thus, this part is the "opposite" of the Pattern Evaluation part in the sense that a pure sequence of Penalties is required to "throw" the LA back into the Pattern Evaluation part again, while a single Reward reconfirms the pattern, returning the LA to the Pattern Acceptance part of the state space.

In other words, the automaton attempts to incorporate past deterministic responses when deciding on a sequence of actions.

We define the “*Ensemble*” characteristic of a set of STPLA as follows: *An event is only signalled to the recipient when all of the STPLA that maintain hypotheses that are consistent with the event, collectively find themselves in the Pattern Evaluation part of the state space.* As soon as one of the STPLA can deterministically⁵ explain an event as being part of the corresponding hypothesized spatio-temporal event pattern, that particular event will be suppressed and no notification will be issued to the recipient.

3 Experiments

In order to evaluate our scheme, we have applied it to both an event simulation system as well as to a real world prototype. This section reports the results obtained using the simulation, while the next section covers the prototype.

Since one of our main aims is handling noisy patterns, we intend to impose “stress” onto our scheme by using a wide range (percentage or degrees) of omission and inclusion errors. We will use q to denote the probability of event omission, while p denotes the probability of event inclusion. We also investigate how the number of states N_1 and N_2 affect the LA’s speed and the accuracy.

As a performance criterion, we have chosen the probability of issuing a notification (alert) when an event takes place. We refer to this probability as P_1 . Intriguingly, when a spatio-temporal pattern produces events, P_1 should be minimized, while when events are novel, P_1 should be maximized. We will presently see that our scheme achieves both. For instance, consider an event that occurs daily, with the possibility, however, that events may get cancelled (causing omission errors). In that case, our scheme should quickly stop alerting the user about these events. In contrast, when novel sporadic events occur, even on a daily basis, our scheme should rather always produce alerts, so that the user is notified about these novel events. Thus, by monitoring our scheme in terms of the index P_1 using various scenarios, we can capture its overall performance.

3.1 Performance after Convergence

Table 1 summarizes the performance after convergence, with a wide range of event inclusion probabilities, p , event omission probabilities, q , *Pattern Evaluation States*, N_1 , and *Pattern Tracking States*, N_2 . The resulting performance is then reported in terms of P_1 , with P_1 being estimated by averaging over 1,000 experiments, each consisting of 100,000 iterations.

In the case of daily patterns, we have varied the omission error probabilities from $q = 0.05$ to $q = 0.2$, thus covering a spectrum of small to high degrees of omission noise. In the case when no patterns are present, we have allowed random encounters to appear with probabilities from $p = 0.05$ to $p = 0.2$.

⁵ The system can easily be generalized for SE and SNE events by rendering the transitions stochastic.

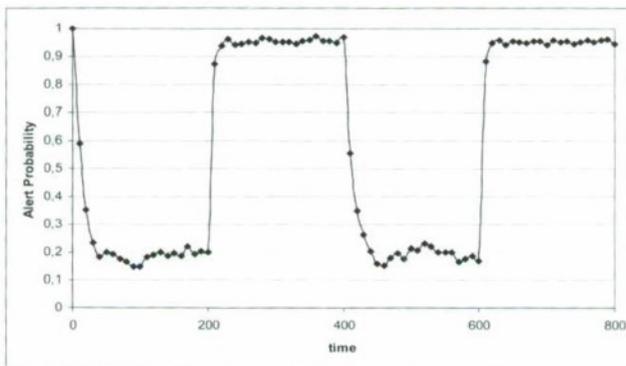
Table 1. Alert probability P_1 under varying conditions

	Daily Pattern			No Underlying Pattern		
	$q = 0.05$	$q = 0.1$	$q = 0.2$	$p = 0.05$	$p = 0.1$	$p = 0.2$
(N_1, N_2)						
(1, 5)	1.5E-8	9.9E-7	6.4E-5	0.735	0.531	0.262
(2, 5)	3.2E-8	2.1E-6	1.4E-4	0.983	0.925	0.680
(3, 5)	4.9E-8	3.3E-6	2.4E-4	0.999	0.992	0.916
(4, 5)	6.7E-8	4.7E-6	3.6E-4	0.999	0.999	0.982
(5, 5)	8.6E-8	6.2E-6	5.2E-4	0.999	0.999	0.996
(5, 4)	1.7E-6	6.2E-5	2.6E-3	0.999	0.999	0.997
(5, 3)	3.4E-5	6.2E-4	0.012	0.999	0.999	0.998
(5, 2)	6.9E-4	6.2E-3	0.062	0.999	0.999	0.998
(5, 1)	0.0137	0.059	0.254	0.999	0.999	0.999

From Table 1, we see that for the best configuration, $N_1 = N_2 = 5$, we get very high accuracy, with the scheme producing a negligible number of superfluous notifications to the user, while alerting the user of almost all novel events, even with high degrees of both omission and inclusion errors.

3.2 Performance in Dynamic Environment

To investigate the ability of our scheme to track spatio-temporal patterns that change with time, we have conducted several experiments in dynamic environments. In all brevity, we report here a representative configuration, where spatio-temporal patterns end after a certain time period, while new ones are introduced every 200th iteration. We modelled this by using an omission error probability

**Fig. 3.** Evolution of the alert probability in a dynamic environment

of $q = 0.2$ when a pattern was present, and with an inclusion error probability of $p = 0.2$ when no pattern was present.

Fig. 3 depicts how the STPLA scheme adapts to the presence and absence of patterns over time. For instance, prior to time instant 200, the probability q was equal to 0.2, implying the presence of a daily pattern. As seen, the STPLA quickly learns to suppress these events, albeit, with some error due to the high omission error probability. When the pattern disappears after 200 time steps, being replaced with novel events only, we observe how quickly the STPLA changes from suppressing the events to alerting the user of them.

We thus conclude by stating that the empirical results confirm the power of STPLA both in noisy and dynamic environments.

4 Prototype

In addition to the empirical results presented in the previous section, we have also implemented a social networking application and conducted real-life tests.

A key requirement of our community based social networking application demands that users can be made aware of the *Presence* of their friends at anytime and anywhere using their mobiles sensing capabilities. The latter requirement is akin to the field of pervasive computing where ad-hoc mode-based architectures are recognized to be a better alternative than infrastructure-based architecture.

We now provide a brief description of our prototype, the details of whose implementation can be found in [14]. Our prototype system consists of two mobile phones: *HTC P3300* and *Sony Ericsson X1*, both of which are equipped with Wi-Fi modules. An ad-hoc network is established to provide a communication platform where our proposed solution for a “Friend Reminder” service runs.

This design is based on the “SmokeScreen” architecture [1], which introduces an effective approach to resolve privacy issues of *Presence Sharing*. The signal generation procedure⁶ referred in [1] is depicted in Fig. 4. However, we have added novel enhancements to the “SmokeScreen” approach, by introducing mechanisms that allow a finer level of privacy control. In brief, we allow the user to specify exactly which of his friends can see the signal of his *Presence*. Accordingly, we let every pair of friends share a symmetric key. This is in contrast to the results presented in [1] where a user shares the information of his *Presence* with his social network at the granularity of his group. A major disadvantage of the latter approach is thus that the user cannot apply a finer privacy control by preventing a specific member of the group from sensing the information of his *Presence* (unless the user does not broadcast the signal of his *Presence*). From a privacy perspective, we believe that the control of the user-related information should be fully under his own control. Thus, every user should be able to authorize the specific people who have the right to reveal his user-related information, and to also isolate other users.

The users must be synchronized to independently update the *Presence* signal and broadcast it periodically. Note that the update is deterministic so that every

⁶ As in [1], we use md5 to compute the signal and sha1 to update the secret key.

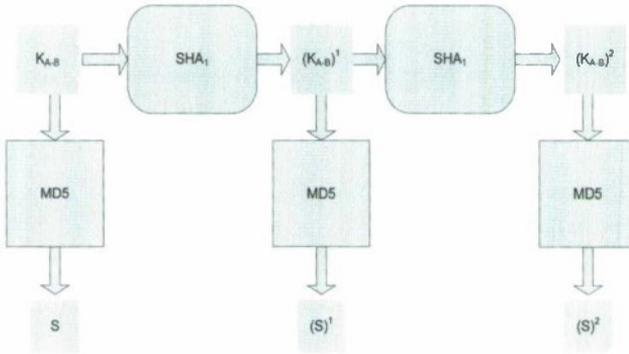


Fig. 4. The signal and key generation over time proposed by [1] and used by us, where k_{A-B} stands for the symmetric key

pair of participating users (for example Alice and Bob) can predict and interpret the time varying broadcast *Presence* signal. The *Presence* signal might vary on the hour and is known only to Alice and Bob, thus preventing impersonation attacks. As alluded to previously, we employed a symmetric key per pair of social contacts. Consequently, the size of the broadcast *Presence* signal increases linearly with the number of social contacts. In order to alleviate this problem, we have used Bloom filters to reduce the size of the *Presence* signal [15], and thus the operation of *Presence* detection reduces to the Bloom filter match operation.

Based on the above architecture, we implemented our STPLA scheme on each mobile phone, allowing suppression of *Presence* notification when the *Presence* is part of a regular pattern. In all brevity, the STPLA scheme made the “Friend Notification Service” less obtrusive by only alerting the user of novel events, but suppressed alerts for regular meetings (e.g., for weekly lectures).

5 Conclusion

In this paper, we have presented the *Spatio-Temporal Pattern Learning Automaton* (STPLA) for the on-line discovery and tracking of patterns in noisy event streams. Our scheme is based on a team of finite automata, rendering it computationally efficient with a minimal memory footprint. The advantages of our approach was demonstrated through extensive simulations, as well as a prototype running on mobile devices. The scheme demonstrated excellent performance under different noise levels and in various dynamic settings. We thus believe the STPLA forms an ideal framework for notification suppression in event notification based systems. As a future work, we intend to formally analyze the behaviour of the STPLA, as well as to extend our prototype to learning interest profiles and adaptive service recommendations.

References

1. Cox, L.P., Dalton, A., Marupadi, V.: SmokeScreen: flexible privacy controls for presence-sharing. In: Proceedings of the 5th International Conference on Mobile Systems, Applications and Services, San Juan, Puerto Rico, pp. 233–245. ACM, New York (2007)
2. Eagle, N., Pentland, A.: Social serendipity: mobilizing social software. *IEEE Pervasive Computing* 4(2), 28–34 (2005)
3. Holmquist, L., Falk, J., Wigström, J.: Supporting group collaboration with interpersonal awareness devices. *Personal and Ubiquitous Computing* 3(1), 13–21 (1999)
4. Bellinger, C.: Simulation and pattern classification for rare and episodic events. Master's thesis, Carleton University, Ottawa, Ontario (2010)
5. Bellinger, C., Oommen, B.J.: On simulating episodic events against a background of noise-like non-episodic events. In: Proceedings of 42nd Summer Computer Simulation Conference, SCSC 2010, Ottawa, Canada, July 11–14 (to appear 2010)
6. Garland, D., Siewiorek, D., Smailagic, A., Steenkiste, P.: Project aura: toward distraction-free pervasive computing. *IEEE Pervasive Computing* 1(2), 22–31 (2002)
7. Youngblood, G., Cook, D.: Data mining for hierarchical model creation. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews* 37(4), 561–572 (2007)
8. Ma, S., Hellerstein, J.: Mining partially periodic event patterns with unknown periods. In: Proceedings of 17th International Conference on Data Engineering, pp. 205–214 (2001)
9. Narendra, K.S., Thathachar, M.A.L.: *Learning Automata: An Introduction*. Prentice-Hall, Englewood Cliffs (1989)
10. Lee, C., Chen, M., Lin, C.: Progressive partition miner: an efficient algorithm for mining general temporal association rules. *IEEE Transactions on Knowledge and Data Engineering* 15(4), 1004–1017 (2003)
11. Thathachar, M.A.L., Sastry, P.S.: *Networks of Learning Automata: Techniques for Online Stochastic Optimization*. Kluwer Academic Publishers, Dordrecht (2004)
12. Tsetlin, M.L.: *Automaton Theory and Modeling of Biological Systems*. Academic Press, London (1973)
13. Narendra, K.S., Thathachar, M.A.L.: *Learning automata: an introduction*. Prentice-Hall, Inc., Englewood Cliffs (1989)
14. Lin, M., Wen, X.: Handling Relations in a Ubiquitous Computing Environments. Master's thesis, University of Agder, Norway (June 2009)
15. Mitzenmacher, M., Broder, A.: Survey: Network applications of bloom filters: A survey. *Internet Mathematics* 1 (2003)

A Graph Model for Clustering Based on Mutual Information

Tetsuya Yoshida

Graduate School of Information Science and Technology,
Hokkaido University
N-14 W-9, Sapporo 060-0814, Japan
yoshida@meme.hokudai.ac.jp

Abstract. We propose a graph model for clustering based on mutual information and show that the clustering problem can be approximated as a combinatorial problem over the proposed graph model. Based on the stationary distribution induced from the problem setting, we propose a function which measures the relevance among data objects. This function enables to represent the entire objects as an edge-weighted graph, where pairs of objects are connected by the edges with their relevance. We show that, in hard assignment, the clustering problem can be approximated as a combinatorial problem over the proposed graph model when data is uniformly distributed. We demonstrate the effectiveness of the proposed approach over the document clustering problem. The results are encouraging and indicate the effectiveness of our approach.

1 Introduction

Clustering is a process of finding a partition of data objects into mutually exclusive and exhaustive groups. The groups are called clusters. The objective is to find clusters of data objects such that data within the same group are similar to each other, while data among different groups are dissimilar. Clustering is a fundamental data processing in various fields, and has been investigated in many research communities, e.g., machine learning, data mining, etc. [6].

In this paper we consider data clustering under the framework in [13], where the clustering problem is formalized as a constrained optimization problem based on mutual information. Since this problem is difficult to solve due to the non-linearity of mutual information and non-convexity of the objective function, several approximation algorithms have been proposed [13,10,9].

Based on the stationary distribution induced from the problem setting, we propose a function which measures the relevance among data objects under the problem setting. Since this function captures the pairwise relation among data objects, the entire data objects can be represented as an edge-weighted graph, where data objects (which correspond to vertices) are connected by edges with their relevance. The edge-weighted graph for the entire data objects is called a data graph in this paper.

We show that, in hard assignment, clustering based on mutual information can be approximated as a combinatorial problem over the proposed data graph when

data is uniformly distributed. Representing the entire data objects as a data graph and formalizing the clustering problem over the graph enable to utilize various graph algorithms to solve the clustering with mutual information. We demonstrate the effectiveness of the proposed approach by utilizing spectral clustering over the proposed graph model and evaluating it on the document clustering problem. The results are encouraging and show the validity of the proposed approach.

Our contributions are: 1) proposal of a graph model for clustering based on mutual information, 2) clarification of the correspondence between the original clustering problem and the combinatorial problem over the graph model, and 3) validation of the proposed approach over the document clustering problem.

Section 2 explains the problem setting. Section 3 explains the details of our approach. Section 4 reports the results of experiments and comparison with other approaches. Section 5 gives concluding remarks.

2 Problem Settings

2.1 Preliminaries

Let \mathbf{X} be a set of data objects. For a set \mathbf{X} , $|\mathbf{X}|$ represents its cardinality.

Suppose X stands for a random variable over the domain \mathcal{X} , and $p_1(x)$ and $p_2(x)$ are probability distributions for X .

Definition 1. *Kullback-Leibler (KL) divergence between two probability distributions $p_1(x)$ and $p_2(x)$ for a random variable X is defined as [1]:*

$$D_{KL}[p_1(x)||p_2(x)] = \sum_x p_1(x) \log \frac{p_1(x)}{p_2(x)} \quad (1)$$

Suppose X and Y are two random variables (their domains are \mathcal{X} and \mathcal{Y}), and $p(x, y)$ stands for their joint probability distribution. Let $p(x)$ and $p(y)$ stand for their marginal probability distributions, and $p(y|x)$ stands for the conditional probability distribution of Y given the observation of X .

Definition 2. *Mutual Information $I(X; Y)$ between two random variables X and Y is defined as:*

$$I(X; Y) = \sum_x \sum_y p(x, y) \log \frac{p(x, y)}{p(x)p(y)} \quad (2)$$

$$= D_{KL}[p(x, y)||p(x)p(y)] \quad (3)$$

2.2 The Information Bottleneck Framework

Data clustering based on mutual information was proposed in [13]. The objective is to find clusters \mathbf{T} of data objects \mathbf{X} such that the clusters are still informative about the specified relevant variable Y . Random variables X and T corresponds to \mathbf{X} and \mathbf{T} , and T should be completely defined given X and irrelevant to Y .

For instance, suppose a set of documents $\mathbf{X}=\{x_1, \dots, x_n\}$ is specified, each of which contains a “bag” of terms to describe the document. Here, the set of

whole terms utilized to describe the documents corresponds to $Y = \{y_1, \dots, y_m\}$. $p(x, y)$ represents the joint probability of a document x containing a term y , and can be estimated by the co-occurrence of x and y . The goal of data clustering is to find a partition $T = \{t_1, \dots, t_k\}$ of X such that T is still informative about Y . Here, each $t \in T$ corresponds to a cluster of documents.

Data clustering is formalized as a constrained optimization problem [13].

Problem 1. Find the conditional probability distribution $p(t|x)$ which minimizes the following objective function

$$\mathcal{L} = I(X;T) - \beta I(T;Y) \tag{4}$$

where $I(X;T)$ and $I(T;Y)$ are mutual information between X and T and between T and Y , respectively. β is a control parameter.

Intuitively, $I(X;T)$ corresponds to the compactness of new representation T for representing the value of X , while $I(T;Y)$ corresponds to the accuracy of T for predicting the value of Y . It was shown that the optimal solution of Problem 1 should satisfy the following self-consistent equations [13,8].

Theorem 1. *When $p(x, y)$ and β are specified, and Markovian relation $T \leftrightarrow X \leftrightarrow Y$ holds, $p(t|x)$ is a stationary point of \mathcal{L} if and only if $p(t|x)$ satisfies the following equations:*

$$p(t|x) = \frac{p(t)}{Z(x, \beta)} \exp(-\beta D_{KL}[p(y|x)||p(y|t)]) \tag{5}$$

$$Z(x, \beta) = \sum_t p(t) \exp(-\beta D_{KL}[p(y|x)||p(y|t)]) \tag{6}$$

2.3 Approximation Algorithms

The closed form formula in eq.(5) indicates that $p(t|x)$ is the stationary distribution under the problem setting. However, $p(t|x)$, the left hand side of eq.(5), implicitly (and non-linearly) affects its right hand side under this framework. Furthermore, the objective function \mathcal{L} in eq.(4) is not convex with respect to $p(t|x)$, $p(t)$, $p(y|t)$ simultaneously. Thus, it is quite difficult to find the global optimal solution of Problem 1.

Several algorithms were proposed to find out approximated solutions of eq.(4) [13,10,9,8]. It is reported that an algorithm called sIB outperformed other algorithms in terms of the quality of clusters. This algorithm returns a hard assignment, i.e., each data is assigned only to one cluster.

3 A Graph-Based Approach

3.1 Preliminaries

A graph $G(V, E)$ consists of a finite set of vertices V , a set of edges E over $V \times V$. The set E can be interpreted as representing a binary relation on V . An edge-weighted graph $G(V, E, W)$ is defined as a graph $G(V, E)$ with the weight on each edge in E . When $|V| = n$, the weights in W can be represented as an

n by n matrix \mathbf{W}^1 , where w_{ij} in \mathbf{W} stands for the weight on the edge for the pair $(v_i, v_j) \in \mathbf{E}$. We set $w_{ij} = 0$ if the pair (v_i, v_j) is not in \mathbf{E} .

3.2 A Pseudo-similarity Function

Based on Theorem 1 and eq.(5), we regard that $D_{KL}[p(y|x)||p(y|t)]$ represents the pseudo-dissimilarity between x (data object) and t (cluster) under the framework in Section 2. Furthermore, we extend this insight from $\mathcal{X} \times \mathcal{T}$ to $\mathcal{X} \times \mathcal{X}$, and propose to utilize KL-divergence as a pseudo-dissimilarity function between data objects for the clustering problem.

Based on the above argument, we propose the following function, which corresponds to a pseudo-similarity function under the framework in Section 2.

Definition 3. A function $s: \mathcal{X} \times \mathcal{X} \rightarrow \mathcal{R}^+$ is defined as

$$s(x_i, x_j) = p(x_j) \exp(-\beta D_{KL}[p(y|x_i)||p(y|x_j)]) \tag{7}$$

where β is the control parameter in Problem 1.

3.3 A Data Graph

The function defined in eq.(7) represents the pairwise relation among data objects. Since a pairwise relation can be represented as a graph, we propose to represent this relation as an edge-weighted graph, using the $s(x_i, x_j)$ in eq.(7) as the weight for the edge (x_i, x_j) .

Definition 4. For a set of objects \mathbf{X} , by mapping each data object to a vertex, an edge-weighted graph $G(\mathbf{V}, \mathbf{E}, \mathbf{W})$ is defined as:

$$\mathbf{V} = \mathbf{X} \tag{8}$$

$$w_{ij} = \begin{cases} s(x_i, x_j) & x_i \neq x_j \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

$$\mathbf{E} = \{(x_i, x_j) | s(x_i, x_j) > 0\} \tag{10}$$

From eq.(8), we abuse the symbol \mathbf{X} to denote the set of vertices in the data graph. Note that the weights are non-negative. We call this graph the **data graph** in this paper. We assume that the data graph G for a given \mathbf{X} is connected².

We define the conditional probability over the data graph³ as

$$p(x_j|x_i) = \frac{w_{ij}}{\sum_j w_{ij}} \tag{11}$$

Proposition 2. The conditional probability in eq.(11) is a stationary distribution in Theorem 1 where $\mathbf{T} = \mathbf{X}$.

¹ A bold italic symbol \mathbf{W} denotes a set, while a bold symbol \mathbf{W} denotes a matrix.

² Each vertex has at least one edge with positive weight. For disconnected graphs, w.l.o.g., each component can be dealt with separately.

³ It is easy to verify that eq.(11) is a valid conditional probability.

Proof. By treating each x_j as t , it is easy to confirm from eqs.(7) and (9). \square

Setting $T = X$ corresponds to one extremal situation where no compression of X is conducted. In that situation, Proposition 2 says that $p(x_j|x_i)$ in eq.(11) over the data graph, based on the function in eq.(7), satisfies the necessary condition for the optimal solution of Problem 1.

3.4 A Graph-Based Formalization

We shall show that, for hard assignment, Problem 1 can be approximated as the following problem over the data graph when data is uniformly distributed.

Problem 2. When the number of clusters k is specified, find k disjoint subsets $\{E_1, \dots, E_k\}$ of edges in the data graph G which minimize

$$J = \sum_{t=1}^k \sum_{w_{ij} \in E_t} w_{ij} \tag{12}$$

and the removal of the edges from G results in k disconnected components.

Objective functions. Note that when random variables X and Y are specified, $I(X; Y)$ is some constant value. Based on this fact, Problem 1 can be transformed into the following equivalent problem for any fixed β (see [13,8]).

Problem 3. Find the conditional probability distribution $p(t|x)$, which minimizes the following objective function

$$F_{TB} = \sum_x \sum_t p(x)p(t|x)(-\log Z(x, \beta)) \tag{13}$$

In the data graph G , the objective function is represented as

$$F_G = \sum_{x_i} \sum_{x_j} p(x_i)p(x_j|x_i)(-\log Z(x_i, \beta)) \tag{14}$$

We define the sum of weights on the edges from x_i as d_i^4 .

$$d_i = \sum_{x_j} w_{ij}, \quad \forall x_i \in X \tag{15}$$

We introduce one assumption to show our result.

Assumption 1. Data is uniformly distributed and $p(x)$ is some constant $c > 0$. Hereafter, Assumption 1 is called as uniform distribution.

Proposition 3. Under uniform distribution, F_G is some constant for X .

⁴ \sum_{x_j} ranges over X and corresponds to \sum_j .

Proof

$$\begin{aligned}
 F_G &= \sum_{x_i} \sum_{x_j} p(x_i)p(x_j|x_i)(-\log Z(x_i, \beta)) \\
 &= c \sum_{x_i} (-\log Z(x_i, \beta)) \sum_{x_j} p(x_j|x_i) \tag{16}
 \end{aligned}$$

$$= c \sum_{x_i} (-\log d_i) \sum_{x_j} \frac{w_{ij}}{d_i} \tag{17}$$

$$= c \sum_{x_i} (-\log d_i) \tag{18}$$

Since $p(x_i) = c$ and $Z(x_i, \beta) = \sum_{x_j} p(x_j) \exp(-\beta D_{KL}[p(y|x_i)||p(y|x_j)]) = \sum_{x_j} w_{ij} = d_i$, and d_i is some constant for each x_i , eq.(16) follows. Eq.(17) follows from eq.(11), and $\sum_{x_j} \frac{w_{ij}}{d_i} = 1$ for each data x_i induces eq.(18). Since each $-\log d_i$ is some constant as in eq.(16), Proposition 3 holds. \square

Compression and cut. Let us consider a 2-way partition of \mathbf{X} into two mutually exclusive and exhaustive sets, *i.e.*, $\mathbf{X} = S \sqcup \bar{S}$ [14]. S and \bar{S} corresponds to two clusters of objects. By removing or cutting the edges between S and \bar{S} , the data graph G is partitioned into two induced subgraphs G_S and $G_{\bar{S}}$ [4], and becomes a (disconnected) graph $\hat{G} = \{G_S, G_{\bar{S}}\}$.

Definition 5. We define the following to characterize a partition.

$$\text{cut}(S, \bar{S}) = \sum_{x_i \in S} \sum_{x_j \in \bar{S}} w_{ij}, \quad \text{cut}(\bar{S}, S) = \sum_{x_i \in \bar{S}} \sum_{x_j \in S} w_{ij} \tag{19}$$

As in eq.(11), for any partition of the data graph G where each induced subgraph G_S with $|S| > 1$, $\sum_{j \in G_S} \frac{w_{ij}}{w_{ij}}$ is a valid conditional probability distribution over G_S .

For each $x_i \in \mathbf{X}$, let us denote the subset of \mathbf{X} which contains x_i as S_i , and the other subset as \bar{S}_i . As in eq.(15), we define the followings.

$$d_{S_i} = \sum_{x_j \in S} w_{ij}, \quad d_{\bar{S}_i} = \sum_{x_j \in \bar{S}} w_{ij} \tag{20}$$

The following relation holds between eqs.(15) and (20) for any x_i in G .

$$d_i = d_{S_i} + d_{\bar{S}_i} \tag{21}$$

We define the conditional probability distribution over $\hat{G} = \{G_S, G_{\bar{S}}\}$ as:

$$\forall x_i \in S, \hat{p}(x_j|x_i) = \begin{cases} \frac{w_{ij}}{d_{S_i}} & x_j \in S \\ 0 & \text{otherwise} \end{cases} \tag{22}$$

$$\forall x_i \in \bar{S}, \hat{p}(x_j|x_i) = \begin{cases} \frac{w_{ij}}{d_{\bar{S}_i}} & x_j \in \bar{S} \\ 0 & \text{otherwise} \end{cases} \tag{23}$$

⁵ \bar{S} is the complement of S .

F_{G_S} and $F_{G_{\bar{S}}}$ are defined as eq.(14), and can be rewritten under uniform distribution as:

$$F_{\hat{G}} = \sum_{x_i} \sum_{x_j} p(x_i)\hat{p}(x_j|x_i)(-\log Z(x_i, \beta)) \tag{24}$$

$$= \sum_{x_i \in S} \sum_{x_j \in S} p(x_i) \frac{w_{ij}}{d_{S_i}} (-\log Z(x_i, \beta)) + \sum_{x_i \in S} \sum_{x_j \in \bar{S}} p(x_i) \frac{w_{ij}}{d_{S_i}} (-\log Z(x_i, \beta)) \tag{25}$$

$$= F_{G_S} + F_{G_{\bar{S}}} \tag{26}$$

Note that $\hat{p}(x_j|x_i)$ defined in eqs.(22) and (23) does not satisfy eq.(5), since $p(S_i|x_i) = 1$ and $p(\bar{S}_i|x_i) = 0$ for all $x_i \in X^6$, and deviates from eq.(5) due to the hard assignment of each x_i into S_i^7 . We would like to minimize the deviation to solve Problem 1. From Proposition 3, minimization of the deviation $F_{\hat{G}} - F_G$ is equivalent to the following problem.

Problem 4. For any set of objects X , find the 2-way partition $X = S \sqcup \bar{S}$ of the data graph G which minimizes $F_{\hat{G}} = F_{G_S} + F_{G_{\bar{S}}}$ in $\hat{G} = \{G_S, G_{\bar{S}}\}$.

Main result. We show the correspondence between Problem 1 and Problem 2. First, we define the following problem.

Problem 5. For the data graph G , find two disjoint subsets $\{E_1, E_2\}$ of edges which minimize

$$J = \sum_{t=1}^2 \sum_{w_{ij} \in E_t} w_{ij} \tag{27}$$

and the removal of the edges from G results in a disconnected graph $\hat{G} = \{G_S, G_{\bar{S}}\}$, where G_S and $G_{\bar{S}}$ are components of \hat{G} .

Claim. In hard assignment, Problem 1 can be approximated as Problem 5 under uniform distribution.

Proof. As explained, Problem 1 can be reduced to Problem 4. Thus, we show the correspondence between Problem 4 and Problem 5. In the following, symbol \Leftrightarrow represents the equivalence, and symbol \simeq represents the approximation.

$$\begin{aligned} \min F_{\hat{G}} &\Leftrightarrow \min \left\{ \sum_{x_i \in S} (-\log d_{S_i}) + \sum_{x_j \in \bar{S}} (-\log d_{S_j}) \right\} \\ &\simeq \min \left\{ \sum_{x_i \in S} d_{S_i} + \sum_{x_j \in \bar{S}} d_{S_j} + \sum_{x_i \in S \cup \bar{S}} (1 - d_i) \right\} \end{aligned} \tag{28}$$

$$\Leftrightarrow \min \left\{ \sum_{x_i \in S} d_{S_i} + \sum_{x_j \in \bar{S}} d_{S_j} \right\} \tag{29}$$

$$\Leftrightarrow \min \{ cut(S, \bar{S}) + cut(\bar{S}, S) \} \tag{30}$$

⁶ S and \bar{S} corresponds to clusters.

⁷ Any hard assignment deviates from eq.(5).

$$\Leftrightarrow \min \sum_{t=1}^2 \sum_{w_{ij} \in \mathbf{E}_t} w_{ij} \quad (31)$$

The first equation holds under uniform distribution from eq.(26). Based on eq.(21), Taylor expansion of log function as $(-\log d_{S_i}) \simeq d_{\bar{S}_i} + (1 - d_i)$ shows that eq.(28) holds. As Proposition 3, since each d_i is some constant in G , it is equivalent to eq.(29). From eq.(20) and the definition of $cut(S, \bar{S})$, eq.(29) is equivalent to eq.(30), and the latter is equivalent to eq.(31). \square

The above *Claim* can be easily extended to more than two clusters.

Claim. In hard assignment, Problem 1 can be approximated as Problem 2 under uniform distribution.

3.5 Clustering Based on Data Graph

Section 3.4 shows that the clustering problem in Section 2 can be tackled by solving the combinatorial problem (Problem 2) over the proposed data graph. Various graph algorithms have been proposed for solving this kind of problem efficiently [11] and can be utilized via the proposed reduction of the problem.

However, it is known that small unbalanced clusters tend to be created under the minimum cut formulation of partitioning [14]. From the objective of data clustering, unbalanced clusters are not desirable. Thus, when solving Problem 2 over the data graph, in addition to minimizing the objective function, it would be important to consider the balance between the clusters.

4 Evaluations

4.1 Application for Document Clustering

Although the proposed method is generic and not specific to document clustering, following the previous work [8,2], we evaluated the proposed approach on the document clustering problem. Similar to the example in Section 2.2, for a given documents \mathbf{X} , the set of terms which are utilized to describe the documents correspond to $\mathbf{Y} = \{y_1, \dots, y_m\}$, and $p(x, y)$ corresponds to the joint probability of a document x and a term y . Since the number of terms are huge in general, the document clustering problem corresponds to the clustering of high-dimensional sparse data. Since the proposed approach is a partitioning based method, we assume that the number of clusters k is specified.

Based on the procedure in [8,2], we evaluated the proposed approach over the 20 Newsgroup data (20NG)⁸, which has been utilized as a standard benchmark in document processing community. Three sets of groups are created, as shown in (Table 1). As in [8,2], 50 documents were sampled from each group in order to create one dataset. We repeated this process and created 10 datasets for each set of groups. For each dataset, we conducted stemming using porter stemmer⁹

⁸ <http://people.csail.mit.edu/jrennie/20Newsgroups/>. 20news-18828 was utilized.

⁹ <http://www.tartarus.org/martin/PorterStemmer>

Table 1. Datasets from 20 Newsgroup dataset

dataset	included groups
Multi5	comp.graphics,rec.motorcycles,rec.sport.baseball, sci.space, talk.politics.mideast
Multi10	alt.atheism, comp.sys.mac.hardware,misc.forsale, rec.autos,rec.sport.hockey, sci.crypt,sci.med, sci.electronics,sci.space,talk.politics.guns
Multi15	alt.atheism, comp.graphics, comp.sys.mac.hardware, misc.forsale, rec.autos, rec.motorcycles, rec.sport.baseball, rec.sport.hockey, sci.crypt, sci.electronics, sci.med, sci.space, talk.politics.guns, talk.politics.mideast, talk.politics.misc

and MontyTagger¹⁰, removed stop words, and selected 2,000 words with large mutual information [1].

4.2 Experimental Settings

Compared Methods. For each dataset, we constructed the data graph in Section 3.3 and conducted clustering by solving Problem 2 over the graph. As described in Section 3.5, it is important to consider the balance among clusters. We utilized spectral clustering to fulfill this objective [14]. For each pair (x_i, x_j) the edges with w_{ij} and w_{ji} are defined in the data graph. However, these should be removed simultaneously for partitioning. Thus, we set the symmetric matrix $(\mathbf{W})_{ij}=(w_{ij} + w_{ji})/2$ in the following experiment.

Two representative normalized graph Laplacian have been proposed and utilized based on the diagonal matrix \mathbf{D} , which is filled with d_i in eq.(15) [14]: $\mathbf{L}_{rw} = \mathbf{I} - \mathbf{D}^{-1}\mathbf{W}$, $\mathbf{L}_{sym} = \mathbf{I} - \mathbf{D}^{-\frac{1}{2}}\mathbf{W}\mathbf{D}^{-\frac{1}{2}}$. We utilized both of them and, constructed \mathbf{H}_{rw} and \mathbf{H}_{sym} . Clustering was conducted on these representations using spherical kmeans (skmeans).

We compared the proposed approach with iIB and sIB in [13,9], and with skmeans [3]¹¹. iIB tries to find the stationary distribution in eq.(5) via projection, and sIB conducts sequential re-assignment of data into clusters. The joint probability $p(x, y)$ was estimated from each dataset using Ristad method [7].

Evaluation Measure. For each dataset, cluster assignment was evaluated w.r.t. the following Normalized Mutual Information (NMI) [12]. Let T, \hat{T} stand for the random variables over the true and assigned clusters. NMI is defined as

$$NMI = \frac{I(\hat{T}; T)}{(H(\hat{T}) + H(T))/2} \quad (\in [0, 1]) \quad (32)$$

where $H(T)$ is Shannon Entropy. The larger NMI is, the better the result is.

Although we have evaluated purity [5], the results are omitted for page limit.

Parameters. β in eq.(7) is the control parameter in the problem setting in Section 2. sIB makes it irrelevant to this parameter by setting it a very large

¹⁰ <http://web.media.mit.edu/~lngo/montytagger>

¹¹ Since skmeans is the standard clustering algorithm for high-dimensional sparse data, and this was used as a baseline method.

value ($\beta=10^4$) [9,8]; however, both iIB and the proposed approach are affected by its value. Thus, we conducted preliminary experiments and set the value as $\beta \in [1, 100]$ for iIB and as $\beta \in [10^{-2}, 1]$ in the following experiments.

The number of eigenvectors l also affects the performance in spectral clustering. Basically it was set as $l = k$ (the number of clusters); however, for Multi5 it was set to 10 since setting l to 5 was too considered as too low.

4.3 Results

We conducted experiments on 30 datasets, 10 for each set of groups. For each dataset we conducted 10 runs of experiment in order to account for the influence of initial configuration in clustering, and calculated their average. However, for sIB, following the procedure in [9,8], for each dataset the best result in 10 runs was utilized to calculate the average. The results are shown in Fig. 1. In Fig. 1, kl-rw (red line) stands for the proposed approach with L_{rw} , and kl-sym (blue line) for the proposed one with L_{sym} . The compared methods are: sIB (green

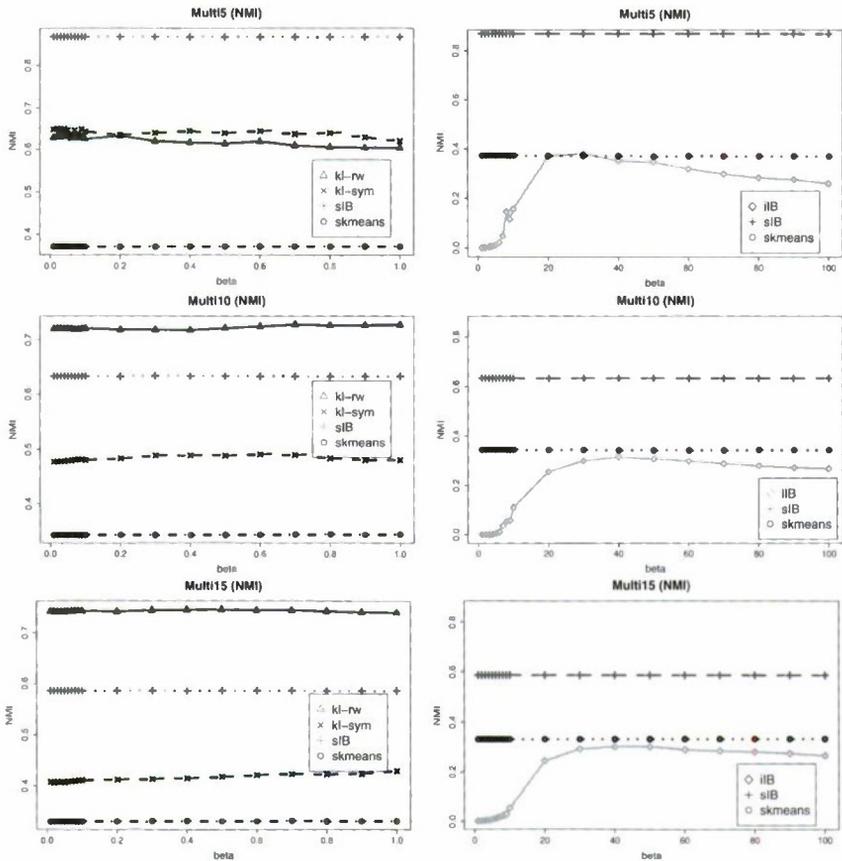


Fig. 1. Result on 20NG (w.r.t. NMI)

line), *ilB* (water blue line), and *skmeans* (black dotted line)). Since β is the main parameter, horizontal axis corresponds to β , and vertical one to NMI.

As for NMI, which corresponds to the correctness of data assignment for clusters, the proposed method with \mathbf{L}_{rw} (*kl-rw*) outperformed other methods with respect to *Multi10* and *Multi15*. On the other hand, for *Multi5*, it outperformed *ilB* and *skmeans*, but it was below *sIB*.

We also compared the proposed approach with the standard spectral clustering [14] using cosine similarity, which is widely utilized in document analysis as a standard similarity measure. Results are summarized in Table 2. Table 2 shows that the proposed approach clearly outperforms the standard spectral clustering. Thus, this validates the effectiveness of the proposed graph model in Section 3.

As for the influence of β , the proposed approach (both *kl-rw* and *kl-sym*) is stable for different values of β and thus can be considered as robust to this parameter. In *ilB*, the performance varied from the value of 1 to 20, but after that it became rather stable with the value of β .

4.4 Discussion

With respect to finding out the stationary distribution in Theorem 1, the proposed approach corresponds to *ilB*. Since the proposed approach outperformed *ilB* in all the datasets in Fig. 1, the results confirmed the validity and the effectiveness of the proposed approach.

The proposed approach formalizes Problem 1 as the corresponding combinatorial problem based on the induced conditional probability over the data graph. \mathbf{L}_{rw} conducts the normalization of graph Laplacian based on the random walk over the graph, which is induced from the weights of the graph [14]. Thus, although both \mathbf{L}_{rw} and \mathbf{L}_{sym} are widely utilized, the former seems to match the proposed approach in terms of the conditional probability interpretation. Furthermore, the results in Fig. 1 also validate that \mathbf{L}_{rw} is more suitable for the proposed data graph. Thus, the proposed approach can be considered as a valid model for data clustering based on mutual information in Section 2.

Although the proposed method is generic and not specific to document clustering, based on the previous work [8,2], we evaluated the proposed approach over the document clustering problem. Since the proposed method (*kl-rw*) outperformed *sIB* for both *Multi10* and *Multi15*, these results showed its effectiveness for the situation where the number of clusters are large. However, although it outperformed the standard spectral clustering, it was below *sIB* for *Multi5*. One of the reasons is that, the original Problem 1 in Section 2 is formalized based on KL divergence, but this divergence can be rather numerically instable when the zero frequency problem in document processing occurs. Coping with this problem is left for future work.

Table 2. Comparison with spectral clustering (NMI)

dataset	\mathbf{L}_{rw}	\mathbf{L}_{sym}	proposal+ \mathbf{L}_{rw} ($\beta = 10^{-2}$)
Multi5	0.573	0.641	0.627
Multi10	0.534	0.497	0.720
Multi15	0.464	0.424	0.741

5 Concluding Remarks

We proposed a graph model for clustering based on mutual information. Based on the stationary distribution induced from the problem setting, a pseudo-similarity function was proposed and utilized to formalize the clustering problem over the proposed graph model. We have shown that, in hard assignment, the clustering problem can be approximated as a combinatorial problem over the proposed graph model when data is uniformly distributed. We demonstrated the effectiveness of the proposed approach by utilizing spectral clustering and evaluating it on the document clustering problem. The results are encouraging and indicate the effectiveness of our approach. We plan to pursue this line of research to overcome the problem related with the instability of KL divergence.

Acknowledgments

This work is partially supported by the grant-in-aid for scientific research (No. 20500123) funded by MEXT, Japan.

References

1. Cover, T., Thomas, J.: *Elements of Information Theory*. Wiley, Chichester (2006)
2. Dhillon, J., Mallela, S., Modha, D.: Information-theoretic co-clustering. In: *KDD 2003*, pp. 89–98 (2003)
3. Dhillon, J., Modha, D.: Concept decompositions for large sparse text data using clustering. *Machine Learning* 42, 143–175 (2001)
4. Diestel, R.: *Graph Theory*. Springer, Heidelberg (2006)
5. Ghosh, J.: Scalable Clustering, pp. 341–364. Lawrence Erlbaum Assoc., Mahwah (2003)
6. Jain, A., Murty, M., Flynn, P.: Data clustering: A review. *ACM Computing Surveys* 31, 264–323 (1999)
7. Ristad, E.: *A Natural Law of Succession*. Technical Report CS-TR-495-95, Princeton University (1995)
8. Slonim, N.: *The Information Bottleneck: Theory and Applications*. PhD thesis, Hebrew University (2002)
9. Slonim, N., Friedman, N., Tishby, N.: Unsupervised Document Classification using Sequential Information Maximization. In: *SIGIR 2002*, pp. 129–136 (2002)
10. Slonim, N., Tishby, N.: Agglomerative Information Bottleneck. In: *Advances in Neural Information Processing Systems (NIPS)*, vol. 12, pp. 617–623 (1999)
11. Stoer, M., Wagner, F.: A Simple Min-Cut Algorithm. *Journal of ACM* 44(4), 585–591 (1997)
12. Strehl, A., Ghosh, J.: Cluster Ensembles - A Knowledge Reuse Framework for Combining Multiple Partitions. *J. Machine Learning Research* 3(3), 583–617 (2002)
13. Tishby, N., Pereira, F., Bialek, W.: The Information Bottleneck Method. In: *Proc. of the 37th Allerton Conference on Communication and Computation*, pp. 368–377 (1999)
14. von Luxburg, U.: A tutorial on spectral clustering. *Statistics and Computing* 17(4), 395–416 (2007)

Shill Bidder Detection for Online Auctions

Tsuyoshi Yoshida and Hayato Ohwada

Department of Industrial Administration, Faculty of Science and Technology,
Research Institute for Science and Technology,
Tokyo University of Science, Japan
yoshida@ohwada-lab.net, ohwada@ia.noda.tus.ac.jp

Abstract. Recently, the online auction has become a popular Internet service. Since the service has been expanded rapidly, security risks in the system remain. Fundamental measures are still required. This paper proposes a method for detecting shill bidders in online auctions. It first detects outliers with a one-class SVM. It then transforms the results into a decision tree using C4.5. The experiment results demonstrate that we can use the resulting rules to classify shill bidders.

Keywords: Online auction, Shill bidders, One-class SVM, Decision tree.

1 Introduction

An online auction is a service that enables ordinary people to sell their items to those who will pay the most for them. Auction sites are set up so that people who wish to sell their items can display them to potential buyers (i.e., bidders). The bidding system enables competition between buyers. The buyer who offers the highest price can acquire the item [1]. Here, both sellers and buyers are ordinary people. They are not professional participants in these auctions.

Recently, online auction services have expanded so rapidly that various security risks in the system have been revealed. Fundamental measures are required against unfair practices. Note that both sides can engage in unfair practices. Both sellers and buyers may suffer from unfair practices. For example, unfair sellers may try to steal money from buyers without sending the purchased products. On the other side, unfair buyers may try to steal items without paying the money. Other types of unfair practices are also observed. Although the bidding systems have to provide a means to protect both types of users (sellers and buyers) from such unfair practices, they always end up reacting after a new type of unfair practice emerges.

Among various unfair practices from the buyer's side, the issue of "shill bidders" [2] remains unsolved. This paper proposes a method to detect shill bidders in an online auction. A characteristic of the proposed method is its semi-automatic function for finding a new type of shill bidders. It first finds outliers based on buyer behavior. It then analyzes the outliers to detect shill bidder behavior. A one-class SVM and decision tree learning algorithm C4.5 are used to find a new type of shill bidders. We demonstrate that a simple combination of these standard learning methods is effective in coping with newly devised unfair practices. Abnormal behavior associated with

unfair practices can be detected in the form of outliers. Rules generated by the decision-tree learning method can classify these outliers and can discriminate unfair practice from innocent outliers. Since buyers in an online auction are ordinary people, innocent outliers do exist. Thus, the one-class SVM alone cannot solve the problem.

This paper is organized as follows. After Section 2 briefly surveys unfair practices in online auctions and related works on efforts to cope with them, Section 3 describes our approach. Section 4 then reports the experiment results. Finally, Section 5 summarizes our findings.

2 Online Auctions and Related Issues

“Shill Bidders” try to get unfair profits by cheating innocent buyers. They try to pull up the price of their items in unfair ways. A typical trick that they use is to employ forged bidders. When a shill bidder goes to sell his item at auction, he begins by putting his item up for auction. He also prepares forged buyers. Typically, forged buyers are actually the shill bidder himself. He uses multiple IDs as buyers. When an innocent buyer places a bid on the item, the forged buyers inflate the price by bidding a higher price. After the price goes up the forged buyers stop bidding, and the cheated innocent has to pay the inflated price.

The automatic bidding support system of online auctions makes the situation worse (Fig. 1). It was originally designed to help innocent buyers. The function of the automatic bidding system is to make successful bids for the items that the buyer wishes to buy. Within a certain price range set by the buyer, the system automatically places bids, inflating the price little by little. A shill bidder can also use this system to create forged buyers. Hence, the use of forged buyers in an online auction is easier than in a traditional auction where a real person has to participate.

To address this problem, a variety of research studies have been conducted. Yokoo et al. point out that this problem is enabled by free mail accounts [3]. A shill bidder can use multiple free mail accounts to imitate the participation of multiple buyers. They also propose an auction protocol that can prevent the participation of forged buyers. Matsuo et al. [4] discuss another auction protocol that can also prevent shill bidders in combination auctions.

The research of Yokoo and Matsuo endeavors to prevent forged buyers, i.e., shill bidders, using the mechanisms of the auction site. This paper seeks to reduce the risk from shill bidders by semi-automatically identifying them. In other words, this paper complements the studies mentioned above.

Deborah sought to predict the closing price for a given auction using the Grey System Theory [5]. Since the number of transactions in an online auction continually increases, the process of monitoring multiple auctions becomes difficult for ordinary buyers. Making the right bid becomes a challenging task for an ordinary bidder. Hence, knowing the closing price of a given auction is an advantage. This information is useful and can be used to ensure a win in a given auction. Our research can provide additional information on the existence of shill bidders that will improve their prediction accuracy.

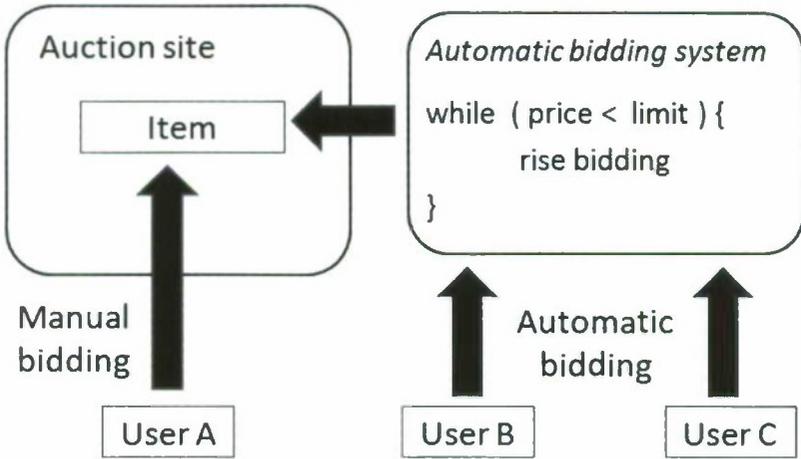


Fig. 1. Automatic bidding and human bidding

3 Semi-automatic Identification of Shill Bidders

In this section we first explain the motivation behind the proposed method and then explain the proposed method in detail.

3.1 Identifying Outliers as Shill Bidders

It is assumed that shill bidders change their tricks every day. Thus, any signature-based method that relies on prior knowledge obtained using a supervised learning method has a problem, since such a method requires manual labeling. A method that can distinguish new tricks automatically is required. To automate the detection of new tricks, we use an unsupervised learning method, namely a one-class SVM. The idea behind this is that shill bidders are outliers and their behavior differs from that of ordinary bidders.

After the one-class SVM distinguishes outliers, the outliers are further analyzed using decision-tree learning method C4.5. Since bidders in an online auction are ordinary people, innocent outliers always exist. To differentiate innocent outliers from shill bidders, we use a manual process to check the results. Data classified into each edge node of the obtained decision tree is examined manually.

We end up modifying the class label of some nodes to "shill bidder" while modifying the class label of other nodes to "innocent outlier." The modified decision tree will be used as the final decision tree for finding shill bidders. Although using C4.5 requires a manual process, the preceding one-class SVM can issue warnings concerning new tricks.

3.2 Details on Finding Shill Bidders

Figure 2 indicates the dataflow inside the auction system and an outline of the proposed method.

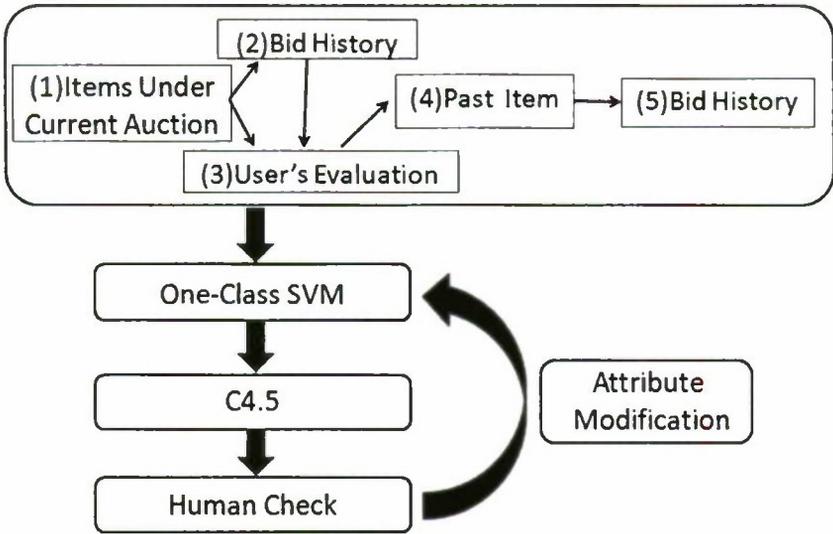


Fig. 2. Proposed method

First, information about the items currently under auction is collected from the auction site (1). The bidding histories of the bidders for the items are then acquired (2). The proposed method also collects the ratings of the bidders identified in the bidding history as well as the bidding history (5) for items that the bidder tried to purchase in earlier auctions. Here, a similarity between bidding histories is the main source of information for confirming the ratings of the bidders.

Table 1. Bidder Attributes

1	User ID of bidder
2	Number of successful bids placed by the bidder during the past three months.
3	Number of times the bidder was rated as “bad” in past auctions.
4	Number of times the bidders were ranked by other participants.
5	Number of participants who ranked the bidder.
6	Ratio of the most frequent party for the bidder.
7	Ratio of the second most frequent party for the bidder
8	Total number of bids made by the bidder during the past three months.
9	Average increase in bids.
10	Average of bidding duration from the preceding bids.
11	Rate at which the amount of an additional bid exceeded 100%.
12	Rate at which the amount of an additional bid was less than 100%
13	Rate at which the amount of an additional bid was exactly 100%

Table 1 lists the attributes acquired through the above process. Attributes 2 through 5 are extracted from the rating information, and attributes 6 through 13 are extracted from bidding histories. Based on these attributes, the one-class SVM extracts outliers. Since these attributes represent bidding histories, i.e., bidders' behavior, the one-class SVM can find bidders whose behavior is abnormal. We use C4.5 to extract human-readable rules for such abnormal bidders. Since abnormal buyers are not always shill bidders, we manually check the found rule to verify that we can interpret the rule to identify shill bidders.

4 Experiment Results

For the experiments, information on 59,949 users and 67,244 items was collected from an auction site [6]. We selected this site to test the proposed method. This site has a basic mechanism for protecting users from shill bidders. For example, bidders have to register their credit card numbers. Credit card information improves the traceability of the transaction. This simple registration process reduces unfair practices. To exclude noise from non-active users, we only analyzed the behavior of users who had participated in auctions more than five times.

4.1 Generated Rule and Classified Buyers

Figure 3 presents the decision tree generated by C4.5. The outliers found by the one-class SVM are classified using the tree in Fig. 3. When the ratio of the outliers was set to be less than 1%, we can define the final tree as the tree that classifies shill bidders from other innocent buyers. The tree in Fig. 3 is generated with an outlier ratio of 0.5%.

In this tree, the branches ending in F nodes with bold outlines seem to classify shill bidders. The branches ending in F nodes with dashed outlines seem to classify active innocent buyers. Although this tree also classifies active buyers as outliers, the interpretation of the end-nodes is not a difficult task for human analysts.

For example, the branch from the root node to the rightmost F node with a bold outline indicates a set of conditions for discriminating shill bidders. Each node in the branch is a condition for the discrimination. It first checks the rate: "(11) the rate at which the amount of an additional bid exceeded 100%" (root node). If the rate is less than or equal to 88.9%, it checks the subsequent conditions, such as "(2) the number of successful bids placed by the bidder during past three months" and "(10) Average of bidding duration from the preceding bids." From this branch, it seems that information such as the bidding duration (10) and the ratio of a second frequent party (11) are important for identifying shill bidders. A short bidding duration (≤ 14.3) seems to indicate the possibility of an automated auction agent, and the ratio of the second most frequent party ($> 41.7\%$, i.e. the buyer has only two parties) seems to indicate the possibility of forged buyers.

In contrast, the branch to the leftmost F node with a dashed outline indicates the conditions, i.e. (11), (2) and (7), for identifying active innocent buyers. Here, the ratio of the second most frequent party ($\leq 7.1\%$, i.e. the buyer has many parties) seems to account for the activity of the buyer.

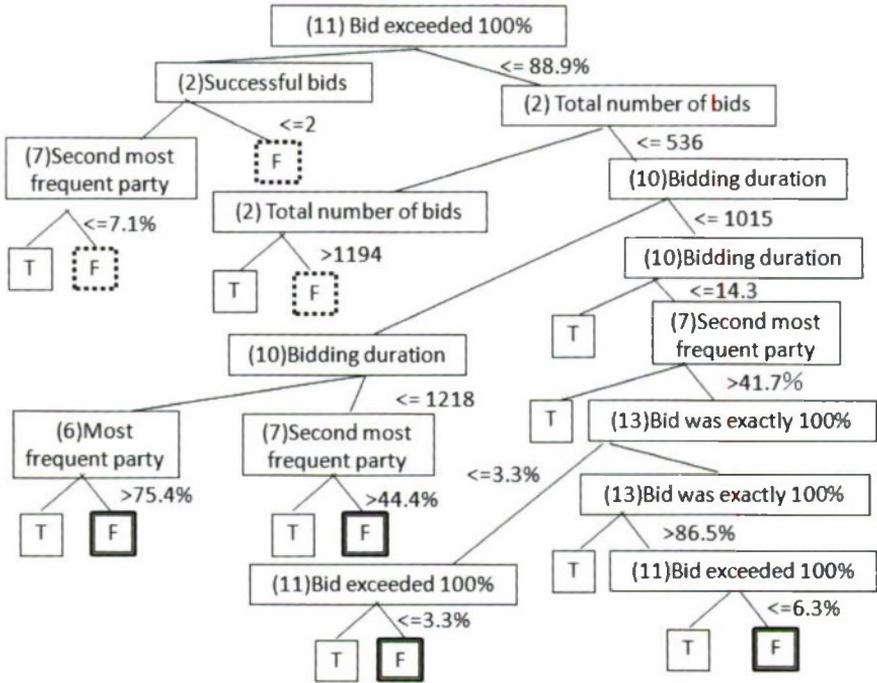


Fig. 3. The rule generated by C4.5

Since the one-class SVM also classifies active innocent buyers as outliers, the tree automatically generated by C4.5 cannot classify shill bidders alone. However, it is difficult to manually check all buyers in an auction without the help of such a tree. Since the numbers of buyers who are classified as outliers is relatively small, the proposed method can make the task of finding shill bidders easier. Moreover, the decision tree also makes interpreting the outlier buyers easier. We can interpret the tree itself to understand the nature of the outliers found.

4.2 Detailed Analysis

Figure 4 plots the change in the number of active buyers and shill bidders found by the proposed method. The horizontal axis represents the ratio of outliers. We change this ratio by changing an input parameter for the one-class SVM program. After the one-class SVM located outliers, a decision tree created by C4.5 was analyzed. The number of active buyers and shill bidders classified by the tree are plotted in this figure. As seen in the graph, the number of shill bidders does not change radically. In contrast, the number of active buyers does change.

Even when we change the ratio of outliers, the group of buyers classified as shill bidders remains relatively stable. Furthermore, 77.5% of buyers who were classified as shill bidders by the proposed method actually were shill bidders.¹ Thus, we believe that the proposed method is useful for identifying shill bidders.

¹ We manually checked all of the buyers who were classified using the proposed method.

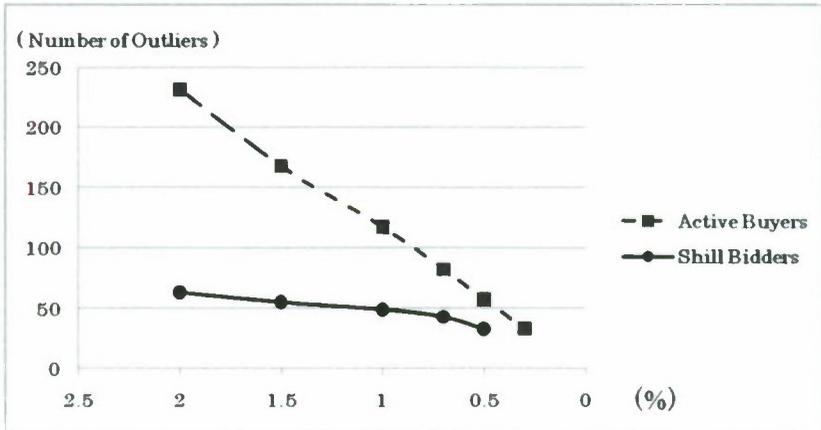


Fig. 4. Detection ratio and breakdown

The percentage of shill bidders is less than we expected. The auction site we analyzed requires an ID, such as a credit card number, in order to create an account. This simple requirement seems to make the registration of forged buyers more difficult and thus contributes to the decrease in the number of shill bidders.

Another important result of our work is a rule for finding active innocent users. The change in the outlier rate seems to control the activity of the identified "innocent active buyers." We can use this result for marketing purposes.

5 Conclusion

This paper proposes a method for detecting "shill bidders" in online auctions. It first detects outliers using a one-class SVM. It then transforms the results into a decision tree using C4.5. The experiment results demonstrate that we can treat the resulting rules as rules for classifying shill bidders. Therefore, the proposed method can in fact detect shill bidders in an online auction. Specific findings of our research are as follows.

1. When the outlier ratio for the one-class SVM is set to around 0.01, our method generates a decision tree that can discriminate shill bidders and active innocent buyers from ordinary buyers.
2. The informative attributes for classifying a shill bidder are the ratio of the most frequent party for the bidder, the ratio of the second most frequent party for the bidder, the average bidding duration from the preceding bids, and the raising rate of any additional bids.
3. The most important feature of the proposed method is its ability to automatically adapt to new shill bidder behavior. The proposed method can classify a shill bidder exhibiting a new behavior as an outlier. The generated tree can help analyze the shill bidder's new behavior.

We can use information about shill bidders for various purposes. For example, the accuracy of the price expectations for future auctions can be improved with this

information. Also, a relative absence of shill bidders can be cited to favorably rate the auction site. Managers of auction sites as well as buyers can use this information to decrease their risk.

References

1. Wolfstetter, E.: Auctions: An Introduction. *Journal of Economic Surveys* 10, 367–420 (2002)
2. Trevathan, J., Read, J.: A Simple Shill Bidding Agent. In: 4th International Conference on Information Technology - New Generations, pp. 933–937 (2007)
3. Yokoo, M., Hirayama, K.: Algorithms for distributed constraint satisfaction: A review. *Autonomous Agents and Multi-Agent Systems. Algorithms for Distributed Constraint Satisfaction* 3(2), 185 (2000)
4. Matsuo, T., Ito, T., Day, R.W., Shintani, T.: A Robust Combinatorial Auction Mechanism based on Detecting Shill Bidders against Fraud Bids. In: The 20th Annual Conference of the Japanese Society for Artificial Intelligence (2006)
5. Lim, D., Anthony, P., Chon, M.H.: Agent for Predicting Online Auction Closing Price in a Simulated Auction Environment. In: Ho, T.-B., Zhou, Z.-H. (eds.) PRICAI 2008. LNCS (LNAI), vol. 5351, pp. 223–234. Springer, Heidelberg (2008)
6. <http://www.bidders.co.jp/>

Mining Hot Clusters of Similar Anomalies for System Management

Dapeng Zhang^{1,3,4}, Fen Lin^{1,3}, Zhongzhi Shi¹, and Heqing Huang²

¹ Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, 100190, China

² Institute of Geographic Sciences and Natural Resources Research, Chinese Academy of Sciences, Beijing, 100101, China

³ Graduate School of the Chinese Academy of Sciences, Beijing, 100039, China

⁴ Institute of Information Science and Engineering, Yanshan University, Qinhuangdao, 066004, China

{zhangdp, linf, shizz}@ics.ict.ac.cn, huanghq@igsnr.ac.cn

Abstract. Recently automatic system management has attracted much attention on mining system log files for anomaly detection, diagnosis and prediction. An important problem in this area is mining hot clusters of similar anomalies for system management. A hot anomaly cluster is defined as a largest-sized group of similar anomalies, whose similarity satisfies some user-specified constraints. While, some major anomalies have common symptoms and are shared by several hot clusters, these clusters do not have to be disjoint. So this problem could not be easily solved by existing clustering algorithms, such as k -means and EM. In this paper we propose a novel heuristic clustering algorithm, named Hot Clustering (HC), for mining these patterns. The key idea of HC is to group neighboring anomalies into hot clusters based on some heuristic rules. To validate our approach, we perform the experiment on bug reports from Bugzilla database by k -means, EM and HC. The experimental results show that our approach is both efficient and effective for this problem.

1 Introduction

Nowadays, computing systems are being increasingly difficult to monitor, manage and maintain. There is an urgent need for automatic and efficient approaches to achieve that [1]. A popular approach for system management is based on analyzing system log files that are stored in structured or unstructured text forms. However, it is costly for system managers to deal with such a large data set. Moreover, log files are generated by a number of different corporate systems, thus the emphasis and wording vary considerably, i.e., anomalies that are truly about the same problem of the system, may be described in different ways by different authors, at varying times and under varying conditions [2]. Thus the effective discovering of hot clusters of similar anomalies for system management constitutes our most urgent problem.

To automatically discover hot anomaly clusters, different types of anomalies must be separated while similar anomalies must be grouped. Thus, we can use

traditional clustering algorithms, which take text reports as input and automatically group each report into a single type. However, as the majority of anomaly clusters have very small size while only a few ones have large size, traditional algorithms are not effective and efficient enough to discover these large clusters with satisfactory similarity. Moreover, since some major anomalies are common symptoms and are shared by several hot clusters, these clusters may be joint with each other. Therefore, this problem could not be easily solved by existing clustering algorithms, such as k -means and EM. It is necessary for us to explore new methods to the detection of hot anomaly clusters.

2 Related Works

Recently automatic system management has attracted much attention on mining system log files for anomaly detection, diagnosis and prediction [1,3,4,5]. One of the key issues is to group similar anomalies in system log files. Tao Li et. al [1] apply text mining techniques to categorize message in log files into common situations, and build an integrated framework of heterogeneous logs for system management. Zhenmin Li et. al [3] classify bug reports into different categories based on text classification and information retrieval techniques. It focuses on investigating impacts of new factors on software errors to improve software design, development, and so on. Mike Chen et. al [4] train decision trees to identify causes of failures from web request logs, thus diagnosing failures in large Internet Sites. Yinglung Liang et.al [5] exploit different classifiers including RIPPER, SVMs and nearest neighbor-based method on event logs from IBM Blue Gene/L, in order to predict failure events of the system.

Different with these works, this paper focuses on mining hot clusters of similar anomalies for system management. Importance of this problem has been enjoying a growing amount of attention. In [6] Srivastava et.al discuss four clustering techniques used for this problem, including k -means, Sammon mapping, EM and Spectral clustering. However, this problem could not be easily solved by these traditional methods. As the anomalies are not uniformly distributed, traditional algorithms are not effective and efficient enough to discover hot anomaly clusters with satisfactory similarity. Moreover, since some major anomalies are common symptoms and are shared by several hot clusters, these clusters may be joint with each other. It can hardly be achieved through traditional algorithms which mainly produce strictly disjoint clusters. Therefore, in this paper we propose a novel clustering algorithm, Hot clustering, which outputs the largest-sized hot anomaly clusters and allows the resultant clusters not to be disjoint.

The proposed algorithm extends classic density-based clustering method with adjustable similarity threshold and multi-class clustering. When no similarity threshold is set and strictly disjoint clusters are required, our algorithm degrades to classic density-based clustering [7,8]. A similar work [9] by Daxin Jiang et. al proposes a density-based hierarchical clustering method to cluster gene expression data. Their algorithm builds a density tree by summarizing clusters and dense areas to explore the cluster structure of a data set, while our approach

groups neighboring anomalies to mine hot clusters of similar anomalies based on some heuristic rules.

3 Problem Specification

This section describes the problem that we focus on, i.e., *given a set of anomalies in a report data set, find the set of hot clusters of similar anomalies*. A report data set is denoted by $D = (x_1, x_2, \dots, x_n)$, containing n anomalies from x_1 to x_n . Each anomaly x_i is represented by a feature vector of length m , where $x_i = (x_{i1}, x_{i2}, \dots, x_{im})$, $i = 1, \dots, n$. A feature represents a keyword to distinguish from different anomalies and group similar anomalies. Feature selection is not only important, but also very challenging, mainly due to the inherent difficulty of the problem as well as the large volume of the text data set [10]. The feature value x_{ij} is determined by TFIDF, which is the Term Frequency and Inverse Document Frequency of the j th feature in the i th document report, where $x_{ij} = TF_i(f_j) \cdot \log(n/DF(f_j))$.

Similarity of any two anomalies x_p and x_q is measured with a distance function, denoted by $d(x_p, x_q)$. Different distance functions can be chosen for different applications. For instance, when using Euclidean distance L_1 , we have $d(x_p, x_q) = \sum_{j=1}^m |x_{pj} - x_{qj}|$. For an anomaly cluster $C = (x_1, x_2, \dots, x_k)$ with k anomalies, we define two measures, *average pairwise distance* and *maximum pairwise distance*, to calculate similarity of the whole cluster.

Definition 1 (Average Pairwise Distance). *Average pairwise distance of an anomaly cluster C is denoted by $ad(C)$:*

$$ad(C) = \frac{\sum_{i=1}^k \sum_{j=1}^k d(x_i, x_j)}{k^2} \quad (1)$$

Definition 2 (Maximum Pairwise Distance). *Maximum pairwise distance of an anomaly cluster C is denoted by $md(C)$:*

$$md(C) = \max d(x_i, x_j), \quad \forall x_i, \forall x_j \in C \quad (2)$$

Average pairwise distance represents the average similarity between anomalies within a cluster, smaller *average pairwise distance* means greater average similarity and more satisfactory hot cluster. *Maximum pairwise distance* reflects the minimum similarity among all anomalies in the cluster, smaller *maximum pairwise distance* means greater minimum similarity and more satisfactory hot cluster.

Hot Clustering aims at finding the largest and most similar anomaly clusters. Since the two goals are incompatible, a tradeoff approach is to find the largest-sized group of anomalies satisfying some user-specified similarity constraints. Definition 3 formally defines a hot anomaly cluster H with three user-specified parameters, *MaxVts*, *MaxDts* and *MinPts*.

Definition 3 (Hot Anomaly Cluster). *A hot anomaly cluster H wrt. $MaxVts$, $MaxDts$ and $MinPts$ is a non-empty subset of D satisfying the following conditions:*

1. $ad(H) \leq MaxVts$
2. $md(H) \leq MaxDts$
3. $\|H\| \geq MinPts$, where $\|H\|$ is the size of H
4. $\forall p \in (D-H)$, if $H^* = (H+p)$, then either $ad(H^*) > MaxVts$ or $md(H^*) > MaxDts$

For a hot anomaly cluster, its average pairwise distance should be no more than $MaxVts$, maximum pairwise distance should be no more than $MaxDts$, and cluster size should be no less than $MinPts$. The last condition is an extension of the third condition, which guarantees that the cluster should contain as many anomalies as possible.

4 A Novel Hot Cluster Discovery Algorithm

4.1 Heuristic Rules

Given the hot cluster parameters $MaxVts$, $MaxDts$ and $MinPts$, a hot anomaly cluster can be discovered in a two-step approach. First, choose an arbitrary anomaly from the data set as a seed. Second, expands it repeatedly until no more anomalies could be added. An important question is how to efficiently expand a seed anomaly to a hot cluster. To this ends, two heuristic rules are designed based on notions of *neighborhood* and *hot degree* respectively.

Definition 4 (Eps-neighborhood). *Eps-neighborhood of an anomaly x_i is denoted by $N_{Eps}(x_i)$:*

$$N_{Eps}(x_i) = \{x_j | x_j \in D \wedge d(x_i, x_j) \leq Eps\} \quad (3)$$

The Eps -neighborhood of an anomaly x_i contains all anomalies within Eps distance away from x_i . It captures the neighbors of an anomaly.

Definition 5 (Hot Degree). *The hot degree of an anomaly cluster C is denoted by $hd(C)$, where $\|C\|$ is the size of C :*

$$hd(C) = \frac{\|C\|}{ad(C)} \quad (4)$$

The hot degree of an anomaly cluster indicates its compactness, which aims at reconciling the two goals of HC. A cluster with higher value of hot degree means it contains more members or has high average similarity. The hot degree of an anomaly x_i , is defined by the hot degree of its Eps -neighborhood in the following equation.

$$hd(x_i) = hd(N_{Eps}(x_i)). \quad (5)$$

Based on these notions, two heuristic rules can be explored:

Neighboring Rule. When expanding a seed anomaly, give priority to its neighbors. In Figure 1(a) for example, x_2 is in the Eps -neighborhood of x_1 , while x_3, x_4 are out of the Eps -neighborhood of x_1 . When Eps is small, we could

still have the Triangle Inequality Theorem hold despite in high-dimensional space, i.e.:

$$d(x_2, x_1) \leq d(x_3, x_1) \leq d(x_3, x_2) + d(x_2, x_1)$$

$$d(x_2, x_1) \leq d(x_4, x_1) \leq d(x_4, x_2) + d(x_2, x_1)$$

If $d(x_3, x_2) < d(x_4, x_2)$, then the establishment of $d(x_3, x_1) < d(x_4, x_1)$ will have a greater probability than $d(x_3, x_1) > d(x_4, x_1)$.

Hot Degree Rule. When choosing seed anomalies for expanding, give priority to anomalies with higher values of hot degree. In Figure 1(b) for example, the *Eps*-neighborhood of x_1, x_2, x_3 , are denoted by N_1, N_2, N_3 respectively. x_2 and x_3 are both in N_1 . Suppose that $hd(x_3) > hd(x_2)$, i.e.:

$$\frac{\|N_3\|}{ad(N_3)} > \frac{\|N_2\|}{ad(N_2)}$$

Suppose that the anomalies in N_1 is very intensive and uniformly distributed, then we have $ad(N_3 \cup N_1) \approx ad(N_3)$, $ad(N_2 \cup N_1) \approx ad(N_2)$. Then we could have

$$\frac{\|N_1 \cup N_3\|}{ad(N_1 \cup N_3)} > \frac{\|N_1 \cup N_2\|}{ad(N_1 \cup N_2)}$$



Fig. 1. Example of heuristic rules

4.2 The Algorithm

The above two heuristic rules form the foundation of the process of hot anomaly cluster discovery. The goal of HC is to gather as many anomalies as possible within a user-specified similarity threshold. A greedy approximation heuristic algorithm is applied which starts from a seed anomaly and then iteratively expands to a hot cluster based on the two heuristic rules. Repeating this process for all anomalies in the data set will generate all hot anomaly clusters.

With respect to the three parameters in Definition 3, *MaxVts* and *MaxDts* are of most important, while *MinPts* does not remarkably affect clustering results when being set in a moderate range. Algorithm 1 also introduces two other parameters of *Eps* and *Pts* to control the searching space. *Eps* is used for neighbor finding, which is set as the maximum distance of acceptable neighbors. *Pts* is used for seed anomaly finding, which is set as the minimum number of neighbors for seed anomalies. All these parameters are detailed and analyzed in section 4.2.

In algorithm 1, Eps , Pts and Eps -neighborhood for all anomalies are initialized first. If the anomaly is a seed anomaly, expand it from its neighborhood, otherwise, ignore it. When expanding a seed anomaly x_i to find a hot cluster, first add all anomalies in its neighborhood to Q_C , in which anomalies are ordered by their hot degree. Second pop the highest hot degree anomaly out of Q_C as the new seed. This step of seed expanding aims at finding neighbor's neighbor for the hot cluster of x_i . This is achieved through a two-step approach. First drop those anomalies if their maximum pairwise distance from the original hot cluster exceeds $MaxDts$. Second, add all left anomalies if the average pairwise distance of the enriched new cluster does not exceed $MaxVts$. All these newly added anomalies will be stored in Q_N , as candidate anomalies for future expanding. All anomalies in the hot anomaly cluster started from x_i will be saved in $N_C(x_i)$. When Q_C is empty but Q_N is not empty, add all the candidate anomalies in Q_N to Q_C . This is achieved by checking the average distance constraint until all possible neighborhoods are examined. As a result, the associated hot clusters of all anomalies in the data set will be extracted.

```

input : Data set  $D$ , Average distance threshold  $MaxVts$ , Maximum pairwise distance
         threshold  $MaxDts$ 
output: a set of hot clusters, for each anomaly  $x_i$ , saved in  $N_C(x_i)$ 

1.1 Initialize neighborhood radius  $Eps$ , seed anomaly density  $Pts$  ;
1.2 Initialize  $N_{Eps}(x_i)$  for all anomalies in  $D$ ;
1.3 for  $x_i$  in  $D$ ,  $i = 1$  to  $n$  do
1.4   if  $\|N_{Eps}(x_i)\| \geq Pts$  then
1.5     Add  $N_{Eps}(x_i)$  to  $N_C(x_i)$ ;
1.6     Add  $N_{Eps}(x_i)$  to  $Q_C$ ;
1.7     Sort  $Q_C$  by anomaly hot degree;
1.8     while  $Q_C$  is not empty do
1.9       Pop  $a$  from  $Q_C$ ;
1.10      if  $\|N_{Eps}(a)\| < Pts$  then
1.11        continue;
1.12      end
1.13       $A_a = N_{Eps}(a) - N_C(x_i)$ ;
1.14      for  $x_j$  in  $A_a$  do
1.15        if  $md(N_C(x_i) + x_j) > MaxDts$  then
1.16          Remove  $x_j$  from  $A_a$ ;
1.17        end
1.18      end
1.19      if  $ad(N_C(x_i) + A_a) \leq MaxVts$  then
1.20        Add  $A_a$  to  $N_C(x_i)$ ;
1.21        Add  $A_a$  to  $Q_N$ ;
1.22      end
1.23      if  $Q_C$  is empty, but  $Q_N$  is not empty then
1.24        Add  $Q_N$  to  $Q_C$ ;
1.25        Empty  $Q_N$ ;
1.26        Sort  $Q_C$  by anomaly hot degree;
1.27      end
1.28    end
1.29  end
1.30 end

```

Algorithm 1. Hot Clustering Algorithm, HC

4.3 Improvements and Complexity

The computational cost of HC can be decomposed into two parts.

1. The time required for neighborhood initialization.
2. The time required for hot cluster extracting.

Neighborhood initialization intuitively need to compare the anomalies one-to-one with the time complexity of $O(n^2)$. This is computationally very expensive, especially for large data set. Thus, we need to partition the data set into several disjoint subsets and initialize the neighborhoods inside these subsets. Algorithm 2 presents the partition method, in which related anomalies are put together to automatically separate far away anomalies into different subsets. There are two parameters for partitioning the data set, *ParVts* and *ParPts*. *ParVts* is the maximum average pairwise distance of the subsets, while *ParPts* is the minimum size of the subsets. The time complexity of Algorithm 2 is $O(nk)$, where k is the number of subsets.

Hot cluster extracting is breadth-first search of the neighborhoods with linear complexity, because only neighbor-reachable anomalies are checked. Thus the most costing computing is actually the calculation of the average pairwise distance. The computing could reuse existing results by Equation 6, where, M is the existing cluster and N is the newly added anomaly set (M and N are disjoint). In the equation, $ad(M)$ is already known, and the newly added set is much smaller than the cluster size ($\|N\| \ll \|M + N\|$). So the time required for distance computing is approximate to $O(\|M\| \cdot \|N\|)$. Additionally, the searching space of a certain anomaly is actually the sum of all neighbor-reachable anomalies from this anomaly. So the time complexity of hot cluster extracting

```

input : Data set  $D$ , Average pairwise distance threshold  $ParVts$ , Density threshold
          $ParPts$ 
output: Disjoint subsets  $P^* = \{P_i\}$ 
2.1 for  $x_i$  in  $D$ ,  $i = 1$  to  $n$  do
2.2   for  $P_j$  in  $P^*$ ,  $i = 1$  to  $m$  do
2.3     if  $ad(P_j + x_i) < ad(S_P + x_i)$  then
2.4        $S_P = P_j$ ;
2.5     end
2.6   end
2.7   if  $ad(S_P + x_i) \leq ParVts$  then
2.8      $Add\ x_i\ to\ S_P$ ;
2.9   end
2.10  else
2.11     $Create\ a\ new\ subset\ P_x\ for\ x_i$ ;
2.12     $Add\ P_x\ to\ P^*$ ;
2.13  end
2.14 end
2.15 for  $P_i$  in  $P^*$ ,  $i = 1$  to  $m$  do
2.16   if  $\|P_i\| < ParPts$  then
2.17      $Remove\ P_i\ from\ P^*$ ;
2.18   end
2.19 end

```

Algorithm 2. Partitioning Data set Based on Average Pairwise Distance

is $O(m.l.d)$, where m is the seed number, l is the size of the sum set and d is the maximum size of the cluster.

$$ad(M + N) = \frac{ad(M).\|M\|^2 + ad(N).\|N\|^2 + 2.D(M, N)}{\|M + N\|^2} \quad (6)$$

To sum up, the time complexity of neighborhood initialization is $O(nk)$, and the time complexity of hot cluster extracting is $O(m.l.d)$. As the majority of anomaly cluster tend to have very small size, i.e., $k \ll n$ and $m \ll n$, the time required for HC is approximate to $O(l.d)$, which are largely determined by neighborhood radius Eps . Thus, the selection of Eps is the key factor affecting the performance of the algorithm, which will be detailed in section 4.3.

5 Experiments

Experiments are performed using bug reports from Bugzilla database [11] to evaluate the proposed algorithm of HC with benchmark algorithms of KM and EM.

5.1 Data Sources

We collected our data from an on-line large open source software project of Mozilla, which contains about thirty-three products, including Calendar, Camino, Composer, Firefox, Thunderbird, Core, Directory, Toolkit, Webtools, websize, etc. Each product has a number of bug reports in Mozilla Bugzilla database. These text reports are individually recorded by tens of thousands of on-line users, including volumes of similar and recurring bugs. They particularly address the problems of the number and similarity of bug clusters. Through hot clustering of similar bugs, system managers can easily triage the anomalies, recognize system brittleness, and gain high level evolutionary information for system development. Although our experiments are performed on bug reports in Bugzilla database, it can be used in other text data sources where high-dimensional clustering need to be applied to discover hot patterns of similar topics from a huge amount of historical data.

Each bug reports in bugzilla databases contains the following attributes of bug ID, summary, time, status, reporter, assignee, severity, bug description, discussion comments, test cases, attachments, and activities, etc. We use all of these attributes except time, status, reporter, assignee, attachments and activities, which are of little relevance for hot anomaly clustering while hard for current automatic analyzing techniques. Our experimental products include addons.mozilla.org, Camino, Calendar and Bugzilla with different data set size and feature size, as shown in Table1.

Table 1. Experimental Data Sets

Product	addons.mozilla.org	Camino	Calendar	Bugzilla
Item Size	2,818	3,790	6,666	12,224
Feature Size	984	1,259	1,694	2,403

5.2 Evaluation Method

One of the most important issues in hot anomaly cluster discovery is the evaluation of clustering results. In general, there are two criteria to investigate cluster validity [12].

1. *Compactness*, the members of each cluster should be as close to each other as possible.

2. *Separation*, the clusters themselves should be widely spaced.

Our study addresses the problem of discovering hot clusters of similar anomalies. Since some major anomalies are common symptoms and are shared by several hot clusters, these clusters do not have to be disjoint. Thus we could only use the compactness criteria to evaluate different hot clustering algorithms. The compactness of a hot anomaly cluster could be described in terms of *Cluster Size*, *Average Pairwise Distance* and *Maximum Pairwise Distance* as follows:

- *Cluster Size*, $\|C\|$, how many members the cluster contains, larger size means more satisfactory hot cluster.

- *Average Pairwise Distance*, $ad(C)$, defined in Definition 1, smaller value means greater average similarity and more satisfactory hot cluster.

- *Maximum Pairwise Distance*, $md(C)$, defined in Definition 2, smaller value means greater minimum similarity and more satisfactory hot cluster.

To investigate the performance of different clustering algorithms, we define two kinds of hot clusters: *largest hot clusters* and *similar hot clusters*. Suppose that the hot cluster set found by KM, EM and HC are denoted by K , E and H respectively, where $K = (K_1, K_2, \dots, K_m)$, $E = (E_1, E_2, \dots, E_n)$, $H = (H_1, H_2, \dots, H_l)$.

- *Largest Hot Clusters*, the group of largest hot clusters is defined by a triple $\langle K_L, E_L, H_L \rangle$, where K_L is the largest cluster in K , E_L is the largest cluster in E , and H_L is the largest cluster in H .

- *Similar Hot Clusters*, a group of similar hot clusters is defined by a triple $\langle K_S, E_S, H_S \rangle$, which should satisfy $\|K_S \cap E_S \cap H_S\| \geq 70\%$, $\min\{\|K_S\|, \|E_S\|, \|H_S\|\}$. That is to say, the similar hot clusters should contain at least 70% same members with the smallest cluster.

5.3 Comparison of the Three Algorithms

The three clustering algorithms are compared over all four products listed in Table 1. All algorithms are implemented in Java and all tests were performed under the same circumstance. Regardless of different distance functions in the clustering algorithm, a uniform distance function is used to measure the similarity of result anomaly clusters. The distance of any two anomalies in the result cluster is computed by $d(x_p, x_q) = \sum_{j=1}^m |B(x_{pj}) - B(x_{qj})|$, where,

$$B(x_{ij}) = \begin{cases} 1, & \text{if feature } f_j \text{ occur in report document } i, \\ 0, & \text{otherwise} \end{cases} \quad (7)$$

The cluster number for KM and EM is got by testing a series of values to obtain the largest and most similar sets of anomalies. And the selected values are 1,400, 2,500, 3,500, and 3,000 for addons.mozilla.org, Camino, Calendar and Bugzilla respectively. Additionally, noisy clusters should be filtered out, whose maximum pair distance beyond 60 or average pairwise distance beyond 35. As for HC, we employ the same value for all products: the neighborhood radius is 8, the maximum pairwise distance is 20, and the average pairwise distance is 10. We eliminate the parameter *MinPts* by setting it to 2 for all products.

The running time for the three clustering algorithms is shown in Figure 2. From Figure 2, it is observed that HC is much faster than the other two clustering algorithms. A possible reason is that, the data set is very intensive in some places, but very sparse in most places. KM and EM need to group all the anomalies into different types, while HC only need to find hot clusters in these intensive places.

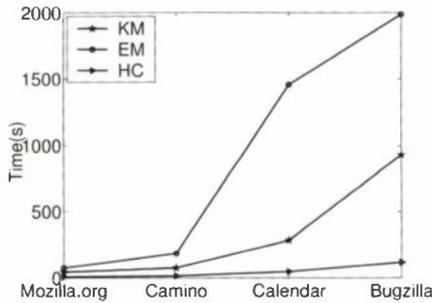


Fig. 2. Running Time of KM, EM and HC

Table 2 presents the evaluation results of the largest hot clusters for the three clustering algorithms of KM, EM and HC. The size of largest hot cluster in HC is close to that in EM, while the average distance and maximum distance are much smaller. Additionally, the size of largest hot cluster in KM is much smaller than the other two algorithms, especially in large data sets. In evaluation of the largest hot clusters, HC outperforms KM and EM on both cluster similarity and cluster size.

Table 3 evaluates the three clustering algorithms with two groups of example similar hot clusters. Examples for addons.mozilla.org show that HC produces

Table 2. Largest Hot Clusters of KM, EM and HC

Product	Algorithm	$\ C\ $	$ad(C)$	$md(C)$	Product	Algorithm	$\ C\ $	$ad(C)$	$md(C)$
addons.mozilla.org	KM	28	25.76	51	Calendar	KM	37	15.29	51
	EM	47	30.28	53		EM	62	13.91	28
	HC	43	9.98	15		HC	136	9.97	20
Camino	KM	11	28.22	56	Bugzilla	KM	26	10.84	24
	EM	25	31.33	52		EM	352	13.73	44
	HC	64	9.94	16		HC	358	10.0	19

much more similar anomaly clusters than KM and EM, while maintaining enough (or even the same number of) members in the cluster. In examples for Camino and Calendar, HC gathers more anomalies than KM and EM, given similar average pairwise distance and maximum pairwise distance. In examples for Bugzilla, HC outperforms KM and EM on both cluster similarity and cluster size.

Table 3. Two Groups of Example Similar Hot Clusters of KM, EM and HC

Product	Group	Algorithm	$\ C\ $	$ad(C)$	$md(C)$	Product	Group	Algorithm	$\ C\ $	$ad(C)$	$md(C)$
addons.mozilla.org	I	KM	8	26.5	51	Calendar	I	KM	7	4.67	9
		EM	6	16.86	35			EM	2	6	6
		HC	4	0	0			HC	7	4.67	9
	II	KM	18	4.14	25		II	KM	3	3.33	4
		EM	17	4.26	25			EM	4	15.67	29
		HC	17	2.06	10			HC	4	7	11
Camino	I	KM	3	9.33	14	Bugzilla	I	KM	6	7.67	13
		EM	2	0	0			EM	6	7.67	13
		HC	3	4	6			HC	5	5.4	7
	II	KM	3	2.67	4		II	KM	5	15	20
		EM	3	2.67	4			EM	9	10.17	23
		HC	6	3.2	8			HC	10	7.2	14

Table 4 shows the result bug reports in the first group of example similar hot clusters in Table 3. Only bug ids and bug summaries are included, and detailed description of a bug report with bug id i can be obtained from https://bugzilla.mozilla.org/show_bug.cgi?id=i. Bug reports found by all three algorithms are included in the row of “*KM, EM, HC intersect*”, while those found by only one or two of the three algorithms are included in the row of the algorithm particular (e.g. “*HC particular*”).

For the similar hot clusters in Table 4, the intersected sets of bug reports describe the same problems with the same summaries. In the example of addons.mozilla.org, both KM and EM find some bug reports different from the problem described by the intersected set, while HC only contains those representing the intersected problem of “Update “Image Zoom” Extension”. Take the bug report 246851 found by KM particularly as an example, though it seems much like the intersected problem, it is in fact about another problem of “Text overlaps badly when zoomed to 200%”. In the example of Camino, EM only finds the problem of “AAHIG - Open Dialog” described by the intersected set, while KM and HC find Bug 188042 and 188041 respectively, whose summary also contain keywords of “AAHIG - Open Dialog”. By in depth analysis of this two particularly found bugs, the one found by KM describes another problem of “add application’s name to open dialog title”, while the one found by HC describes the same problem as the intersected one, which is about “support document preview and multiple selection”. For the example of Calendar, HC performs as good as KM, and EM misses five bug reports. In the example of Bugzilla, similar as the example of addons.mozilla.org, KM and EM improperly find bug 297791 which is obviously different from the problem described by the intersected set. These four examples prove that HC can find more accurate and larger size of hot clusters of similar bug reports than KM and EM.

Table 4. Resultant Bug Reports in the first group of Similar Hot Clusters

Product	Algorithm	BugId	Bug Summary	Product	Algorithm	BugId	Bug Summary		
addons. mozilla. org	KM	251210	Update "Image Zoom" Extension	Calendar	KM,EM, HC <i>intersect</i>	325295	crash if I close the mail window while checking for new mail		
	EM	254074				341607			
	HC	258661				348422			
	<i>intersect</i>		258662					376313	
		KM	246851		Update is not friendly to text zoom (200%)	<i>particular</i>	KM	335899	crash if I close the mail window while checking for new mail
		EM	249413		Add imagezoom extension		HC	338525	
		HC	285749		The Image Zoom ...			341607	
	347716	New links under ...		348422					
EM	<i>particular</i>	249413	Add imagezoom extension			376313			
		285749	The Image Zoom ...						
Camino	KM,EM,HC <i>intersect</i>	187773	AAHIG - Open Dialog	Bugzilla	<i>intersect</i>	KM	313122	implement validations and database persistence functions	
		187776				EM	313123		
	KM	188042	AAHIG - Open Dialog title as "Navigator Open"			HC	313125		
	HC	188041	AAHIG-Open Dialog support multiple selection				313126		
	<i>particular</i>					313129			
					KM,EM <i>paticular</i>	297791	All instances should have ...		

6 Conclusions

In this paper, we formulate the problem of mining hot clusters of similar anomalies for system management. We show that this is not an easily-solved problem by the existing clustering algorithms. We propose a new heuristic density-based algorithm HC to solve this problem. The key idea of HC is to group neighboring anomalies into hot clusters based on some heuristic rules. The experimental result show that our approach is robust, more efficient and effective than k -means and EM for this problem. We believe that the HC algorithm will greatly help the system management.

Acknowledgements

This work is supported by the Natural Science Foundation of China (No. 60775035, 60970088, 60933004, 60903141), National Basic Research Priorities Programme (No. 2007CB311004), and National Science and Technology Support Plan (No. 2006BAC08B06).

References

1. Peng, W., Li, T., Ma, S.: Mining logs files for computing system management. In: ICAC 2005, Seattle, WA, USA, pp. 309-310 (2005)
2. Topol, B., Ogle, D., Pierson, D., Thoensen, J., Sweitzer, J., Chow, M., Hoffmann, M.A., Durham, P., Telford, R., Sleth, S., Studwell, T.: Automating problem determination: A first step toward self-healing computing systems. In: IBM White Paper (October 2003)
3. Li, Z., Tan, L., Wang, X., Lu, S., Zhou, Y., Zhai, C.: Have things changed now? an empirical study of bug characteristics in modern open source software. In: ASID 2006, San Jose, California, USA, pp. 25-33 (2006)

4. Chen, M.Y., Zheng, A.X., Lloyd, J., Jordan, M.I., Brewer, E.A.: Failure diagnosis using decision trees. In: ICAC 2004, New York, NY, USA, pp. 36–43 (2004)
5. Liang, Y., Zhang, Y., Xiong, H., Sahoo, R., Sivasubramanian, A.: Failure prediction in ibm bluegene/l event logs. In: ICDM 2007, Omaha, Nebraska, USA, pp. 583–588 (2007)
6. Srivastava, A.N., Zane-Ulman, B.: Enabling the discovery of recurring anomalies in aerospace problem reports using high-dimensional clustering techniques. In: IEEE Aerospace Conference 2006, p. 17 (2006)
7. Ester, M., Kriegel, H.-P., Sander, J., Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: KDD 1996, Portland, Oregon, USA, pp. 226–231 (1996)
8. Hinneburg, A., Keim, D.A.: An efficient approach to clustering in large multimedia database with noise. In: KDD 1998, New York, NY, USA, pp. 58–65 (1998)
9. Jiang, D., Pei, J., Zhang, A.: Dhc: A density-based hierarchical clustering method for time series gene expression data. In: BIBE 2003, Bethesda, MD, USA, pp. 393–400 (2003)
10. Forman, G.: An extensive empirical study of feature selection metrics for text classification. *Journal of Machine Learning Research* 3, 1289–1305 (2003)
11. Mozilla.org Bugzilla (2005), <https://bugzilla.mozilla.org>
12. Halkidi, M., Batistakis, Y., Vazirgiannis, M.: On clustering validation techniques. *Journal of Intelligent Information Systems* 17(2-3), 107–145 (2001)

A Stratified Model for Short-Term Prediction of Time Series

Yihao Zhang¹, Mehmet A. Orgun¹, Rohan Baxter², and Weiqiang Lin²

¹ Department of Computing,
Maequarie University, Sydney, NSW 2109, Australia
{yihao.zhang,mehmet.orgun}@mq.edu.au

² Australian Taxation Office,
Canberra ACT 2601, Australia
{rohan.baxter,wei.lin}@ato.gov.au

Abstract. This paper develops a model for short-term prediction of time series based on Element Oriented Analysis (EOA). The EOA model represents nonlinear changes in a time series as strata and uses these in developing a predictive model. The strata features used by the EOA model have the potential to improve its forecasting performance on nonlinear data relative to the performance of existing methods. We demonstrate the characteristics of the EOA model using an empirical study of stock indices from eight major stock markets. The study provides comparisons of the accuracy and time efficiency between ARIMA, Neural Networks and the EOA model. Our findings indicate that the EOA model is a promising approach for short-term time series prediction.

Keywords: Short-term Prediction, Time Series, Element Oriented Analysis.

1 Introduction

Short-term prediction in time series has had significant practical applications across different domains in recent years. For example, Wild [18] contributed a method for accurate short-term forecasting of traffic volume time series. His approach achieved satisfactory predictions of traffic volumes at road intersections. Darbellay and Slama [4] forecast short-term electricity demand using existing time-series methods. Furthermore, Gorr and his colleagues [7] extended the topic to the short-term forecasting of crimes. Their results provide a novel approach with applications in the prevention of potential crimes.

Many methods for short-term time series prediction have been reported in the academic literature. Those methods include Exponential Smoothing [16], GARCH [2], ARIMA [1] and Neural Networks [8], to name a few. In terms of practical implementations and applications in the time series domain, ARIMA and Neural Networks are considered to be two mainstream models for short-term prediction [5],[15].

The technical limitations of the ARIMA and Neural Network models have been discussed in the literature [4],[6],[10],[14],[19]. A key limitation of ARIMA

models is that the assumptions of linear and stationary time series are insufficient in many real-world applications [6]. Furthermore, it is a difficult task to build an ARIMA model because it requires good domain knowledge in the specific area of its application [10].

Unlike ARIMA models, Neural Networks do not have the linearity assumption[6]. However, Neural Networks can also be difficult to configure (and train) successfully. Darbellay and Slama [4] have concluded that Neural Networks had no established procedure for identifying the optimal network structures. A related issue is that Neural Networks have the potential of overfitting leading to inaccurate predictions[13]. Therefore, the application of Neural Networks often involves a potentially time consuming empirical trial-and-error approach to obtain an accurate prediction [4].

The main idea behind our stratified predictive model is to discover and represent the dynamical nonlinear changes in a given time series and utilize them to assist forecasting through using Element Oriented Analysis (EOA) proposed by Zhang et al [21]. In this paper, we investigate if EOA based stratified model improves short-term prediction performance relative to the linear ARIMA models while requiring considerably less training effort than the application of Neural Networks in time series analysis.

The rest of the paper is organized as follows. Section 2 introduces the basic ideas behind the EOA model based on a simple example of time series. Then we address the framework of the EOA model and building of the strata for time series prediction in Section 3. In Section 4, the relative performance of the EOA model is demonstrated by an experimental study on the time series indices from eight major stock exchange markets. In particular, we compare the accuracy and time efficiency between the results of the ARIMA model, Neural Networks and the EOA model. The last section summarises the paper's findings and discusses future work directions.

2 Element Oriented Analysis

EOA is a methodology for developing predictive models and not an algorithm. The EOA methodology involves the design of new features or attributes based on a segmentation of the original data. The initial idea of EOA has been proposed and partially used to predict corporate bankruptcy by Zhang et al [21]; we omit a detailed explanation of the EOA model in this paper. In that application, the data was segmented and the segment characteristics were used to add new informative features.

In the time series application reported in this paper, the time series training data is segmented into strata. Informative features are then extracted from these strata and used in an AutoRegression model. The most critical aspect of the application of EOA to time series prediction is how to choose the elements. The following example gives the definition of two elements that we will use later.

Those elements are said to represent a latent relationship within a given dataset in terms of certain intrinsic properties.

Example 1. Suppose that a given dataset consists of ten observations with one binary target variable (Y) and one numeric explanatory variable (X) as follows:

$$\begin{aligned}
 Y &: 0, 1, 1, 0, 1, 1, 0, 0, 1, 0, \\
 X &: 7, 3, 4, 6, 3, 4, 7, 6, 4, 7,
 \end{aligned}$$

Suppose that the study objective from the analysis of this dataset is to find the relationship explaining what kind of X is more likely to cause the case of either $Y = 0$ or $Y = 1$. According to Definition 1, some intrinsic properties are extracted into the new informative features. One of the simple ways to do this is to discover the horizontal and vertical intrinsic properties within the dataset as shown in Figure 1.

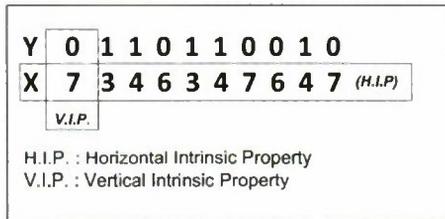


Fig. 1. Two Intrinsic Properties in the Dataset

In the figure, the vertical intrinsic property refers to the variance of X in terms of the 0/1 change of Y . In addition, the horizontal intrinsic property shows the status of each single data point within the entire X that should be discovered. Therefore, we define two elements. Element s_1 represents the probability of explanatory variable given the occurrence of the target variable Y . We use the following function (1) to describe the element:

$$s_1 = P(X | Y = 1) \quad \text{or} \quad s_1 = P(X | Y = 0) \tag{1}$$

where $P(X | Y)$ is the conditional probability of X when Y occurs.

Another element s_2 is used to depict the dataset from the viewpoint of the observations across all attributes. As a result, s_2 states the possibility of the overall partition for the observation between $Y = 1$ and $Y = 0$. For example, we might use a clustering algorithm based on a distance matrix to calculate the belongingness possibility by the following function (2)

$$s_2 = \frac{1}{\sum_{j=1}^2 \frac{d_1}{d_j}} \quad \text{or} \quad s_2 = \frac{1}{\sum_{j=1}^2 \frac{d_2}{d_j}} \tag{2}$$

where d_1 and d_2 represent the distance from the observation to cluster for $Y = 0$ and $Y = 1$. According to functions (2) and (3), we obtain two elements to replace the original explanatory variables as shown below

$$\begin{array}{l} Y : \quad 0, \quad 1, \quad 1, \quad 0, \quad 1, \quad 1, \quad 0, \quad 0, \quad 1, \quad 0 \\ s_1 : \quad 0.6, \quad 0, \quad 0, \quad 0.4, \quad 0, \quad 0, \quad 0.6, \quad 0.4, \quad 0, \quad 0.6 \\ s_2 : \quad 0.11, \quad 0.86, \quad 0.87, \quad 0.2, \quad 0.86, \quad 0.87, \quad 0.11, \quad 0.2, \quad 0.87, \quad 0.11 \end{array}$$

In the above example, the two elements s_1 and s_2 are said to reveal the latent structure between the target variable (Y) and the original explanatory variable (X). Furthermore, the two elements contain the same number of observations as the original dataset and express the original data in terms of either the view of attributes or the whole dataset. Therefore, they are named as *Structure Elements*. The defined elements are chosen based on insights and knowledge of the intended application. These two elements mentioned in above example use segments or strata.

We next describe how the elements are used to do the predictions. In general, Element Oriented Analysis (EOA) methodology has the following components:

1. New elements representing the informative features are generated by segmenting the original dataset.
2. The resulting model uses the new elements (and optionally the original data).
3. The resulting model is multi-level using a Local-Global hierarchy resulting from the use of new Elements based on segments and original data.

Here, the term Local-Global hierarchy refers to two steps of Element Oriented Analysis. Step 1 is Local Level (LL), for determining the elements for each individual application. Step 2 is Global Level (GL), which uses the Elements from LL to meet the modeling objective.

EOA has been applied to other applications such as modeling a classifier predicting whether a credit card holder is good or bad. In this paper, the modeling objective is the prediction of the next k -values (for a small k) in a time series. A more detailed explanation about how EOA works on prediction modeling is given in the following Section 3.

One practical concern is that the same dataset could be segmented into elements in many different ways, which depends on the data domain and intended applications of the model. Therefore, an important part of the application of the EOA model is the discovery and design of the elements. In some cases, especially in time series prediction problems, the difference between two successive data points is a key part of the original data. Therefore, an element may be defined to state this change and combined with the original explanatory variables to predict the future values. Due to the fact that this kind of an element differs from the Structural Element conceptually, we call it the Changing Element. The Changing Element (CE) must also contain the same number of observations as the original data. The idea of a change element was also discussed in [20] in a hierarchical distribution method for extracting knowledge from temporal health records.

In the following section, we focus on how to design an efficient stratified model for accurate time series prediction.

3 Time Series Prediction by EOA

A time series exhibits changes through time. We apply EOA to time series applications by the design of a Changing Element (CE) that describes the nonlinear change from time series at the Local Level (LL). This CE is then used in the Global Level (GL) to estimate a prediction function.

To simplify the presentation, we firstly assume that $\{Y_t\}$ is a time series following a general Autoregressive process,

$$Y_t = \frac{\mu}{\phi(B)}, t = 1, \dots, n. \tag{3}$$

where n is the number of observations for the time series, $\phi(B) = 1 - \phi_1 B - \dots - \phi_p B^p$ is a polynomial in B of degree p and B is the backshift operator. For example, $BY_t = Y_{t-1}$. μ is a constant. In the presence of nonlinear changes, the time series might be affected by unobserved events. To describe the actual time series $\{X_t\}$ subject to the influence of nonlinear changes, the following model is considered:

$$X_t = Y_t + f(t), t = 1, \dots, n. \tag{4}$$

where Y_t follows a general Autoregressive process described in function (3). $f(t)$ is a parametric function that represents the nonlinear change of the actual time series X_t . According to function (3), we obtain a new expression of X_t

$$X_t = \frac{\mu}{\phi(B)} + f(t), t = 1, \dots, n. \tag{5}$$

Then function (4) is converted to:

$$X_t = F(t)\phi^{-1}(B), t = 1, \dots, n. \tag{6}$$

where $F(t) = \phi(B)f(t) + \mu$. We now select $F(t)$ as the CE of time series X_t . Since μ is a constant, $F(t)$ is specified as follows:

$$F(t) = \alpha(B)f(t) \tag{7}$$

where $\alpha(B) = 1 + \mu - \alpha_1 B - \dots - \alpha_s B^s$ is a polynomial in B of degree s , and $f(t)$ is called the Changing Element Function (CEF). If the time series has a linear assumption, the CEF $f(t)$ is followed by the general Moving Average process:

$$f(t) = \frac{\epsilon_t}{\beta(B)}, t = 1, \dots, n. \tag{8}$$

where n is the number of observations for the original observed series. $\beta(B) = 1 - \beta_1 B - \dots - \beta_q B^q$ is a polynomial in B of degree q . And the CE ϵ_t is a sequence of white noise random variables with zero mean and variance σ_ϵ^2 . If the

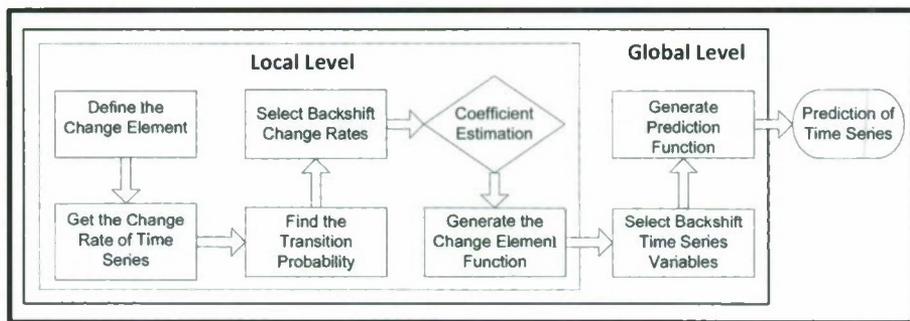


Fig. 2. The Framework of the Stratified Predictive Model based on EOA

time series has no assumption of linearity and it has a sufficient length to train, we might consider the CEF $f(t)$ as follows:

$$f(t) = \frac{\theta(B)}{\alpha(B)} \Delta, t = 1, \dots, n. \tag{9}$$

where $\theta(B) = 1 - \theta_1 B - \dots - \theta_r B^r$ is a polynomial in B of degree r . Δ is CE representing nonlinear variance of time series. To minimize the overfitting, we apply conditional transition probability into Δ , which explains the probability of the states of continuously increasing, continuously decreasing or fluctuation, etc. Note that the details of conditional transition probability Δ are given in section 3.1.

According to Definition 3, the EOA based model should have LL and GL in time series prediction. Therefore, we design three steps in LL and two steps in GL. The comprehensive framework of our stratified predictive model is shown in Figure 2.

To better explain our EOA based stratified model, we first assume a time series $X = \{x_1, x_2, \dots, x_n\}$. According to function (7) and function (9), the goal in LL is to find CE and CEF from X and the goal in GL is to find a Prediction Function based on function (5). In the following, we discuss the EOA model in more detail.

3.1 Finding Changing Element Function in Local Level

As mentioned in previous paragraph, CE can be chosen in many ways representing a change between time series points. However, the selection of an optimal CE is beyond the scope of this paper, although we might consider it in our future work. In this study, we consider the transition probability Δ to be the CE and detected in LL. According to the framework in Figure 2, the following three steps are designed to find the CE and CEF.

Step 1. Obtain the Observational Sequence. We first generate a series of change rates cr_t in (10) to describe the change of the original time series $X = \{x_1, x_2, \dots, x_n\}$.

$$cr_t = (x_t - x_{t-1}) / x_{t-1}, (2 \leq t \leq n). \tag{10}$$

Here the change rate cr_t is formed by calculating between every two consecutive time series data x_t and x_{t-1} . Here t 's are the same time points as in the original series, and n is the length of the observed series. We now obtain a change rate sequence (11) from the original time series:

$$C = \{cr_1, cr_2, \dots, cr_{n-1}\}. \tag{11}$$

Then another new sequence (12) is created to express the possible difference between two consecutive change rates as follows.

$$\delta = \{\delta_1 = (cr_2 - cr_1), \dots, \delta_{n-2} = (cr_{n-1} - cr_{n-2})\} \tag{12}$$

Step 2. Find the CE Δ . From the sequence (12), three states can be defined to represent whether the change rate increases, decreases or keeps the same. We define that S_s represents that the new value is the same as the prior one; S_u represents that the new value is stronger compared with the prior one (the value has increased); and S_d represents that the new value is weaker compared with the prior one (the value has decreased). Accordingly, any difference between the change rates can be represented by these three states.

$$\delta_i \in (S_s, S_u, S_d). \tag{13}$$

Now, we find the CE of transition probability shown in Figure 3.

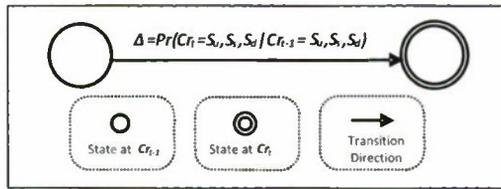


Fig. 3. The Changing Element of Transition Probability

There are nine values among S_s , S_u and S_d for every two consecutive δ_i and δ_{i-1} . These nine values are denoted by $\{s_1, s_2, \dots, s_9\}$ representing the CE Δ of transition probability, where $s_k = P(\delta_i | \delta_{i-1}), (1 \leq i \leq (m - 2), 1 \leq k \leq 9)$.

Step 3. Generate CEF. The obtained CE of transition probability is applied in LL to further estimate the CEF. According to function (6), we consider the explanatory variables from the change rate sequence (10) and the CE Δ in this step. The CEF is estimated as function (14)

$$cr_t = \beta_0 + \sum_{i=1}^n \beta_i cr_{t-i} \Delta. \tag{14}$$

where β_0 is an intercept and β_i represent the coefficients in CEF. Here i is a certain interval between cr_t and another observation. In order to guarantee the best fit of the CEF, we also need to select significant explanatory variables in terms of a predefined significant level. Usually, the significant level 0.01 is adopted as the selection threshold.

3.2 Finding a Prediction Function in Global Level

In the Global Level of the EOA model, the CEF is applied to build a prediction function through another two steps described as follows

Step 1. Shift the CEF into the Prediction Function. The CEF $f(t)$ accurately explains the nonlinear change trend of time series. According to function (4), it therefore is allocated to be a replacement of constant μ in the prediction function. All possible lag variables are regarded to be independent variables in the initial prediction function. In addition, 0.01 significant level is adopted as the selection threshold.

Step 2. Find the Prediction Function. The selection from Step 1 is performed to all independent variables and is repeated until all trivial independent variables are filtered. These selected independent variables and the CEF $f(t)$ are eventually formed into the prediction function (15) as follows.

$$x_t = \alpha_0 + \sum_{i=1}^n \alpha_{1i} x_{t-i} + \sum_{i=1}^n \alpha_{2i} cr_{t-i}. \quad (15)$$

where α_0 is an intercept and α_{1i} and α_{2i} represent the coefficients in the prediction function, i is a certain lag interval between the different observations.

Our stratified predictive model not only provides a solution to overcome the linear assumption from ARIMA by using nonlinear CE Δ , but gives an established function to predict short-term time series. Due to the fact that EOA model adopts autoregression as the main body, we only need to estimate several parameters in the prediction function in the Global Level. As a result, the risk of overfitting is much lower than with Neural Networks. In addition, the training and design time efficiency of the EOA model should be better than that of Neural Networks as well. In the next section, we compare the prediction performance of the EOA model with ARIMA and Neural Networks through a real world empirical study.

4 Empirical Study

In the following, we first discuss the selected time series and the experimental setup. Then we provide the time series prediction results along with discussion.

4.1 Experiment Data and Setup

The selected time series are daily stock indexes from eight major stock exchange markets. The eight markets are stock exchange markets of United States of America, United Kingdom (from 02/Jan/1986 to 31/Dec/2004), Canada, Germany (from 02/Jan/1986 to 30/Dec/2004), Japan (from 04 /Jan/1988 to 30/Dec/2004), Spain (from 30/Dec/1991 to 29/Dec/2004), Taiwan (from 24/Jan/1989 to 31/Dec/2004) and Singapore (from 08/Jan/1988 to 31/Dec/2004).

According to the reported successful applications, such as [3], [4], [5], [11], [10], [13], [15], [17], we choose ARIMA and Neural Network as two benchmarks in this experiment.

For the ARIMA approach, we adopt the viewpoint of Man [9], who specified the order $P = 2$ of the autoregressive model in addition to the order $Q = 2$ of the moving average model which can predict the best result. For the Neural Networks approach, we cite the research of Nam and Schaefer [12] who obtained an accurate prediction of international airline passengers by applying BPNN (Back-Propagation Neural Network). We have made many comparisons between the different structures of BPNN based on their successful experience. In the end, we notice that BPNN with three layers and twelve hidden nodes outperforms the other structures for these time series. Therefore, we run three prediction models, BPNN (3, 12, 1), ARIMA (2, 2) and ours for each index time series respectively in this experiments. We also record the prediction values and the corresponding running time.

Predictive accuracy is the most important performance criterion in this application[19], so we report two frequently used predictive accuracy measures, the Mean Absolute Percentage Error (MAPE) and the Mean Squared Error (MSE) in our comparison.

In our stratified model, the CE presents the correct change trend of the time series if the accuracy of prediction is satisfied. As a result, we need to train the time series until the residual of the prediction falls into a certain range. In this experiment, we set -0.1 to 0.1 to be acceptable range. Figure 4 shows an example

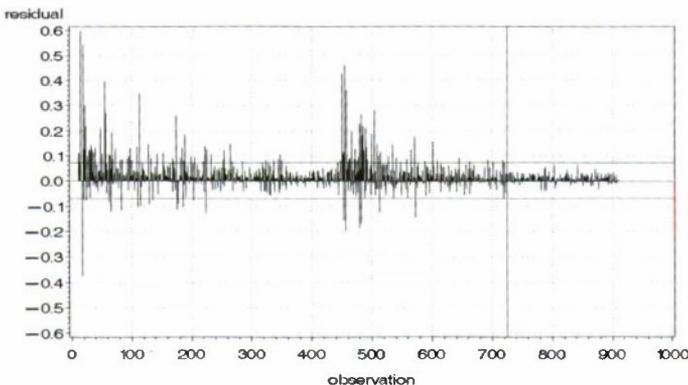


Fig. 4. An Example of Training Time Series in USA index

of training USA index time series. In this example, the prediction result from our stratified model has the fluctuations during day 1 to day 600. These fluctuations shrink from day 600 and stay stable in the range -0.1 to 0.1 until day 725. Hence, we select the first 725 days as the training set for the USA index.

4.2 Results

Tables 1 and 2 record the one-day ahead prediction results among BPNN, ARIMA and EOA over ten consecutive trading days and twenty consecutive trading days.

Table 3 records the running time of three methods in the tests of 1 day, 10 days and 20 days.

In the 10 trading days test (Table 1), the EOA model outperforms BPNN and ARIMA for five index time series (USA, UK, Taiwan, Germany and Canada index time series). BPNN performs slightly better for Spain, Singapore and Japan index time series.

However, we observe from Table 3 that BPNN requires much more computing time than our model to obtain an accurate prediction. The main reason is that Neural networks need more time to train on the time series. For example, it consumes 20.63 seconds to predict 10 trading days in Spain index time series

Table 1. Prediction Accuracy of Three Methods over 10 Trading Days

	This Work		BPNN		ARIMA	
	MAPE	MSE	MAPE	MSE	MAPE	MSE
USA	3.44%	1.76407E-06	4.70%	2.61984E-06	9.80%	9.71E-06
UK	4.12%	4.69458E-06	5.67%	8.22354E-06	10.55%	2.88696E-05
Taiwan	5.40%	0.001526522	5.61%	0.001593115	9.99%	0.004888072
Spain	7.18%	2.62739E-06	7.16%	2.50998E-06	16.16%	1.69405E-05
Singapore	3.33%	2.40069E-05	2.80%	1.91522E-05	7.75%	0.00013073
Japan	0.64%	2.09492E-05	0.54%	1.41529E-05	0.62%	2.21725E-05
Germany	2.12%	7.53072E-06	2.33%	9.99303E-06	3.15%	2.06237E-05
Canada	5.25%	3.97333E-06	6.39%	6.40492E-06	11.47%	0.000018844

Table 2. Prediction Accuracy of Three Methods over 20 Trading Days

	This Work		BPNN		ARIMA	
	MAPE	MSE	MAPE	MSE	MAPE	MSE
USA	4.34%	3.23746E-06	4.52%	3.23885E-06	8.16%	8.76E-06
UK	3.74%	4.7031E-06	4.44%	5.85332E-06	8.90%	2.23537E-05
Taiwan	3.70%	0.000852496	4.27%	0.001045931	6.97%	0.002903487
Spain	7.08%	4.54583E-06	8.32%	6.58587E-06	15.60%	2.28342E-05
Singapore	2.99%	2.0497E-05	2.47%	1.5324E-05	6.30%	0.000091764
Japan	0.63%	2.29944E-05	0.55%	1.60741E-05	0.98%	6.16616E-05
Germany	2.19%	9.0955E-06	2.19%	9.65026E-06	3.35%	2.36186E-05
Canada	4.17%	2.92397E-06	5.28%	4.74761E-06	9.72%	0.000014404

Table 3. Running Time (in second) of Three Methods

	This Work			BPNN			ARIMA		
	1 day	10 days	20 days	1 day	10 days	20 days	1 day	10 days	20 days
USA	0.13	1.33	2.63	2.08	20.83	41.63	0.08	0.83	1.63
UK	0.14	1.43	2.83	2.12	21.23	42.43	0.12	1.23	2.43
Taiwan	0.11	1.13	2.23	2.13	21.33	42.63	0.11	1.13	2.23
Spain	0.16	1.63	3.23	2.06	20.63	41.23	0.05	0.53	1.03
Singapore	0.11	1.13	2.23	2.01	20.13	40.23	0.11	1.13	2.23
Japan	0.12	1.23	2.43	2.05	20.53	41.03	0.12	1.23	2.43
Germany	0.13	1.33	2.63	2.06	20.63	41.23	0.12	1.23	2.43
Canada	0.11	1.13	2.23	1.99	19.93	39.83	0.05	0.53	1.03

10 days running time = output time + 1 day running time×10
 20 days running time = output time + 1 day running time×20
 output time =0.03 second

while our model spends 1.63 seconds only. In 20 trading days test (Table 2), the EOA model has the best accuracy for six time series (USA, UK, Taiwan, Spain, Germany and Canada index time series). BPNN does better for the Singapore and Japan index time series. Therefore, we might conclude from the experiment that our model is competitive in both accuracy and time efficiency.

5 Conclusions

In this paper, we have propose the Element Oriented Analysis model to predict short-term time series. The EOA model mitigates some technical drawbacks of ARIMA and Neural Networks. The accuracy and time efficiency of the EOA model relative to ARIMA and Neural Networks is demonstrated by an experiment on stock indexes. Comparing with these two mainstream models, the experimental results suggest that the EOA based stratified model is competitive in accuracy and time efficiency. Our further work will extend the EOA model to work on other real-world financial applications, such as credit scoring and bankruptcy prediction.

Acknowledgements

This work has been supported in part under the Australian Research Council’s Linkage Projects Funding Scheme (project number LP0561985).

References

1. Box, G., Jenkins, G., Reinsel, G.: Time Series Analysis: Forecasting and Control, 3rd edn. Prentice Hall, Englewood Cliffs (1994)
2. Brook, C., Burke, S.P., Persaud, G.: Benchmark and the accuracy of garch model estimation. International Journal of Forecasting 17, 45–56 (2003)

3. Cottrell, M., Girard, B., Girard, Y., Mangeas, M., Muller, C.: Neural modeling for time series: A statistical stepwise method for weight elimination. *IEEE Transactions on Neural Networks* 6, 1355–1364 (1995)
4. Darbellay, G., Slama, M.: Forecasting the short-term demand for electricity - do neural networks stand a better chance? *International Journal of Forecasting* 16, 71–83 (2000)
5. Faraway, J.: Time series forecasting with neural networks: A comparative study using the airline data. *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 47, 231–250 (1998)
6. Gooijer, J., Hyndman, R.: Twenty-five years of time series forecasting. *International Journal of Forecasting* 22, 443–473 (2006)
7. Gorr, W., Olligschlaeger, A., Thompson, Y.: Short-term forecasting of crime. *International Journal of Forecasting* 19, 579–594 (2003)
8. Hippert, H.S., Pedreira, C.E., Souza, R.C.: Neural networks for short-term load forecasting: A review and evaluation. *IEEE Transactions on Power Systems* 16(1), 44–55 (2001)
9. Man, K.S.: Long memory time series and short term forecasts. *International Journal of Forecasting* 19, 477–491 (2003)
10. Melard, G., Pasteels, J.M.: Automatic arima modelling including interventions, using time series expert software. *International Journal of Forecasting* 16, 497–508 (2000)
11. Monica, A., Fred, C.: How effective are neural networks at forecasting and prediction? a review and evaluation. *International Journal of Forecasting* 17, 481–495 (1998)
12. Nam, K., Schaefer, T.: Forecasting international airline passenger traffic using neural networks. *Logistics and Transportation* 31, 239–251 (1995)
13. Nguyen, H., Chan, W.: Multiple neural networks for a long term time series forecast. *Neural Comput. Appl.* 13(1), 90–98 (2004)
14. Poskitt, D.S.: On the specification of cointegrated autoregressive moving-average forecast system. *International Journal of Forecasting* 19, 503–519 (2003)
15. Tang, Z., Almeida, C., Fishwick, P.: Time series forecasting using neural networks vs. box-jenkins methodology. *Simulation* 57, 303–310 (1991)
16. Taylor, J.W.: Exponential smoothing with a damped multiplicative trend. *International Journal of Forecasting* 19, 273–289 (2003)
17. Weigend, A.S., Huberman, B.A., Rumelhart, D.E.: Predicting Sunspots and Exchange Rates with Connectionist Networks. In: *Nonlinear Modeling and Forecasting*, pp. 395–432. Addison-Wesley, Reading (1992)
18. Wild, D.: Short-term forecasting based on a transformation and classification of traffic volume time series. *International Journal of Forecasting* 13, 63–72 (1997)
19. Zhang, G., Patuwo, B.E., Hu, M.Y.: Forecasting with artificial neural networks: the state of the art. *International Journal of Forecasting* 14, 35–62 (1998)
20. Zhang, Y., Orgun, M.A., Lin, W., Graco, W.: An application of time-changing feature selection. In: Williams, G.J., Simoff, S.J. (eds.) *Data Mining. LNCS (LNAI)*, vol. 3755, pp. 203–217. Springer, Heidelberg (2006)
21. Zhang, Y., Orgun, M.A., Lin, W., Graco, W.: Mining multidimensional data through element oriented analysis. In: Ho, T.-B., Zhou, Z.-H. (eds.) *PRICAI 2008. LNCS (LNAI)*, vol. 5351, pp. 556–567. Springer, Heidelberg (2008)

Using ASP to Improve the Information Reuse in Mechanical Assembly Sequence Planning*

Lingzhong Zhao, Xuesong Wang, Junyan Qian, and Tianlong Gu

School of Computer Science and Engineering, Guilin University of Electronic Technology,
Guilin, 541004 China

{zhaolingzhong, wangxuesong, qjy2000, cctlgu}@guet.edu.cn

Abstract. Modern product design and manufacturing process are highly integrated and exposed to frequent changes. This has made information reuse play an increasingly important role in improving the efficiency of the product development process. Mechanical Assembly Sequence Planning (MASP) is a key issue in the manufacturing of a product. Known methods for MASP are not satisfactory from the aspect of information reuse. This paper proposes an Answer Set Programming (ASP) based solution to MASP, where information reuse is enhanced by dividing an ASP program into EDB (extensional database) and IDB (intensional database) such that IDB can be shared by all the assemblies with the same number of parts. Compared with other approaches for MASP, this is a great advantage. Experiments are conducted to show the applicability and performance of our method by using different answer set solvers.

Keywords: Mechanical Assembly Sequence Planning, Answer Set Programming, EDB, IDB.

1 Introduction

The highly competitive nature of global market of manufacturing products has made product design and manufacturing strategies integrated, computerized and always exposed to frequent changes. In this environment, information reuse in these two processes plays an increasingly important role in improving the efficiency of the product development process. Mechanical Assembly Sequence Planning (MASP) is the task of finding the feasible or optimal sequence that puts the initially separated parts of an assembly together to form the assembled product. A MASP algorithm takes as input the CAD model of an assembly produced by the product design process and produces feasible assembly sequences for the assembly. Much effort has been devoted to this research and many methodologies have been proposed. In literature, there exist a large number of algorithms or systems for assembly sequence generation. These systems differ both in the representation of assembly sequences and in the reasoning technique used to identify feasible sequences. Classic methods include

* This work is supported by National Natural Science Foundation of China (No.60803033, 60663005, 60903079, and 60963010), and Guangxi Natural Science Foundation of China (GuiKeQing 0728093, 0728089, and 0542036).

interactive systems, which work by asking user questions to obtain information necessary to construct feasible assembly sequences [1,17], and cut-set methods that use cut set algorithm to find feasible assembly sequences [7]. The planning algorithm based on OBDDs is a variation of the classic methods [6], and can be viewed as an attempt to attack the combinatorial state explosion problem in storing all feasible assembly sequences. Another line of research is to use soft computing techniques, such as simulated annealing algorithm and genetic algorithm to generate assembly sequences [9, 12]. There also exists an effort to build expert knowledge based systems to generate feasible assembly sequences [16,18].

The above mentioned assembly sequence generation systems and algorithms are not satisfactory from the aspect of information reuse. For example, in the OBDD (Ordered Binary Decision Diagram) based method proposed in [6], once the liaison graph or the interference relation changes, all OBDDs describing the contact and inference information of the original assembly have to be rewritten for the new assembly. This process is very time-consuming and usually requires expert knowledge. In contrast, there is significant information reuse in expert knowledge based systems. But the reused knowledge is related to the special structures of assemblies [18]. This paper is, however, mainly concerned with the reuse of geometric-based knowledge of assemblies.

MASP is a special kind of planning problem. In the last decade, an important method for solving planning problems is to make use of declarative programming languages, such as Answer Set Programming (ASP), to make the solution to be declarative [11]. For example, an ASP based method allows us to divide the planning process into two stages: problem description in ASP and using general purpose answer set solvers, such as DLV, smodels and emodels, to find solutions [4,10,13]. The main advantage of a declarative method is that it allows professionals to be concerned mainly with "what" a solution must satisfy and not with the details "how" to find a solution to the problem. From our point of view, this separation provides a chance of information reuse. Specifically, what a solution must satisfy can be divided into two parts: one part is case-sensitive, and the other is applicable to a class of problem. The latter part is the information that can be reused. If we store these information with a logic program, this division corresponds naturally to the concepts of EDB (extensional database), which represents a collection of facts, and IDB (intensional database), which represents the reasoning components [14].

ASP is a kind of logic programming language under answer sets semantics. The expressiveness, declarative nature and existence of efficient answer set solvers has made ASP a mainstream tool for knowledge representation and reasoning [2]. This paper proposes an ASP based method for MASP, where all assembly knowledge is represented with ASP rules. The case-sensitive information, such as contact and interference relation is included in EDB, and general information that is applicable to a class of problem cases is included in IDB. When the case-sensitive information is changed, we only change the EDB of the knowledge, and IDB is left unchanged. It is shown that the information reuse is greatly improved in this method; and acceptable performance is also achieved.

2 Preliminaries

We first briefly introduce ASP [3]. The answer set semantics of logic programs treats a rule with variables as shorthand for the set of its ground instances. So in defining the answer sets semantics we assume that all the rules in a program do not contain variables. We follow the terminology style of DLV and write classic negation as “~” [10]. Let A be an atom, a literal takes the form A or $\sim A$, where A is a *positive literal* and $\sim A$ is a *negative literal*; A and $\sim A$ are called *complementary literals*.

An *extended disjunctive logic program* P is a set of rules, and each rule r is of the form:

$$L_1 \vee \dots \vee L_k :- L_{k+1}, \dots, L_m \text{ not } L_{m+1}, \dots, \text{not } L_n$$

where $n \geq m \geq k \geq 0$, each L_i is a literal, and *not* is the negation as failure (NAF). We define $head(r) = \{L_1, \dots, L_k\}$ as the head of r , $pos(r) = \{L_1, \dots, L_m\}$ and $neg(r) = \{L_{m+1}, \dots, L_n\}$ as the positive and negative literals present in body of r , respectively. In particular, a rule r without head is called a *constraint*.

Next is the definition of the answer sets for extended logic programs without NAF. Let π be an extended logic program without *not*, and *lit* be the set of ground literals in the language of π . An answer set for π is any minimal subset S of *lit* such that

- a) for each rule $r \in \pi$, if $pos(r) \subseteq S$, then there exists some $l \in head(r)$ such that $l \in S$;
- b) if S contains complementary literals, then $S = lit$.

This definition can be extended to programs with NAFs as follows. Let π be an extended logic program, and *lit* be the set of ground literals in the language of π . For any set $S \subseteq lit$, let π^S be the program obtained from π as follows

$$\pi^S = \{r' \mid r \in \pi, neg(r) \cap S = \emptyset, head(r') = head(r), pos(r') = pos(r), neg(r') = \emptyset\}.$$

Clearly π^S does not contain *not*, so its answer sets are already defined. If S is one of them, then S is an answer set for π .

Assembly Sequence Planning [5,8]

A *mechanical assembly* is a composition of interconnected parts forming a stable unit. Each part is a solid rigid object, that is, its shape remains unchanged. Parts are interconnected whenever they have one or more compatible surfaces in contact. Surface contacts between parts reduce the degree of freedom for relative motion.

A *subassembly* is a nonempty subset of parts that either has one element (i.e., only one part) or is such that every part has at least one surface contact with another part in the subset. Although there are cases where it is possible to join a pair of parts in more than one way, unique assembly geometry will be assumed for each pair of parts.

It is assumed that whenever a subassembly is formed, all connections between its parts are established. Therefore a subassembly can be characterized by its set of parts. Given two subassemblies characterized by their sets of parts S_1 and S_2 , joining S_1 and S_2 is an *assembly task* if $S = S_1 \cup S_2$ is a subassembly. An assembly task is said to be *geometrically feasible* if there is a collision-free path to bring the two subassemblies into contact from a situation in which they are far apart. For the purpose of verifying the geometric feasibility of an assembly task, Gottipolu and Ghosh introduced a

translation function T from the viewpoint of disassembling [5]. In this paper we will redefine the function from the viewpoint of assembling. In specific, T is defined as

$$T: P \times D \times P \rightarrow \{0,1\},$$

where P is the set of parts of an assembly, and $D = \{1,2,3,4,5,6\}$ denoting the six directions (1, 2, 3 for X+, Y+, Z+, and 4, 5, 6 for X-, Y- and Z-, respectively). $T(a, d, b) = 1$ if and only if part b has the freedom of translational motion w.r.t. part a in direction d from far away.

Example: The value of the translation function for the assembly in Fig.1 is 1 on the following set of triples:

{(a,3,b) (a,4,b) (a,6,b) (a,1,c) (a,3,c) (a,4,c) (a,5,c) (a,6,c) (a,5,d) (b,1,a) (b,3,a)(b,6,a) (b,1,c) (b,3,c) (b,4,c) (b,5,c) (b,6,c) (b,5,d) (c,1,a) (c,2,a) (c,3,a) (c,4,a)(c,6,a) (c,1,b) (c,2,b) (c,3,b) (c,4,b) (c,6,b)(c,5,d) (d,2,a)(d,2,b)(d,2,c)}.

The *assembly process* consists of a succession of assembly tasks, each of which consists of joining subassemblies to form a larger subassembly. The process starts with all parts separated and ends with all parts properly joined to form the whole assembly. It is assumed that exactly two subassemblies are joined at each assembly task, and that after parts have been put together, they remain together until the end of the assembly process.

It is also assumed that whenever two parts are joined all contacts between them are established. Due to this assumption an assembly can be represented as an undirected graph $\langle P, C \rangle$, where P is the set of nodes and C is the set of edges. Each node in P corresponds to a part in the assembly and there is an edge in C connecting every pair of nodes whose corresponding parts have at least one surface contact. The elements in C are referred to as *connections*, and $\langle P, C \rangle$ is referred to as the *assembly's connection graph*. A connection encompasses all contacts between parts.

Example: Fig.1 gives an assembly in its exploded view (a) and assembled view (b). The connection graph for the assembly is shown in Fig.1(e).

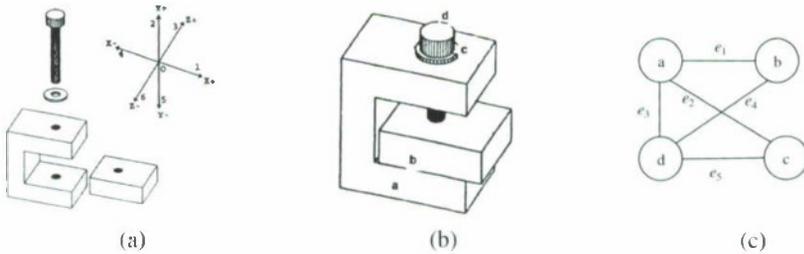


Fig. 1. An example assembly in its (a) exploded view and (b) assembled view, and its corresponding (c) connection graph

The *state* of the assembly process is the configuration of the parts at the beginning or at the end of an assembly task. The configuration is given by the contacts that have been established. Since whenever two parts are joined all contacts are established, the configuration is given by the connections that have been established.

For an assembly with n parts, an *assembly sequence* is an ordered set of $n-1$ tasks. Alternatively, assembly sequences can be represented by an ordered sequence of states [8]. In this paper we will use an ordered list of binary vectors to denote an assembly sequence, where each vector corresponds to a state and the number of the list elements is equal to the number of parts.

Example: Since there are five connections for the assembly in Fig.1, we can use a vector (b_1, \dots, b_5) to denote an assembly state where connection e_i is established only if $b_i=1$. Then an example assembly sequence is:

$$(00000) - (10000) - (11000) - (11111) \quad (*)$$

which corresponds to an assembly sequence where connection e_1 is established firstly, e_2 is established secondly, and e_3, e_4, e_5 are established thirdly.

A *linear assembly sequence* is one in which each task involves the insertion of a single part into the other subassembly [15]. An assembly for which a linear assembly sequence exists is called a linear assembly. This paper is mainly concerned with linear assemblies. By definition each linear assembly sequence may correspond to one or more orders of the assembly parts. For example, the assembly sequence above is obviously linear, from which we could get two assembly orders:

$$\{abcd, baed\}.$$

Next we show that from any element of the set of orders, the sequence (*) can be reproduced. We first define the state corresponding to a non-empty prefix of an assembly order. Let (b_1, \dots, b_n) denote a state where there are n connections in total, and b_i indicates whether the connection b_i is established. The map f is defined as:

$$f(a_1 \dots a_k) = (b_1, \dots, b_n), \quad (k \leq n)$$

where $b_i=1$ if and only if connection e_i is established in the subassembly $\{a_1, \dots, a_k\}$.

Take the assembly order $abcd$ as an example. By definition we have:

$$f(a) = (00000), f(ab) = (10000), f(abc) = (11000), f(abcd) = (11111).$$

The above states exactly correspond to the sequence (*). So in linear assembly, the space of assembly sequences can always be reproduced from the set of possible assembly orders. Hence we can safely use the assembly order of parts to represent the corresponding assembly sequence.

An assembly sequence is *geometrically feasible* if all its assembly tasks are geometrically feasible. To verify the geometric feasibility of an assembly sequence, two types of constraints, connectivity constraints and precedence constraints, must be considered [5]. The *connectivity constraints* specify which parts are connected to other parts in terms of an assembly operation. In specific, we require that:

*Any part assembled at a step should have some contact
with at least one previously assembled part.*

The *precedence constraints* represent the fact that some assembly tasks have to be implemented before the others, otherwise, they will interfere with latter assembly operations. In specific, we require that:

For any part assembled at a step, there exists a collision-free path to bring the part into contact with the subassembly that consists of previously assembled parts.

Example: The assembly sequence "acdb" is not feasible because there exist no direction in which part b can be brought into proper contact with the subassembly that consists of a, c and d.

3 Formulation of Assembly Knowledge

This section will discuss the description of assembly knowledge in terms of DLV language.

3.1 Representation of Assembly Sequences

Following the discussion in section 2, we will use assembly orders or permutations of assembly parts to represent assembly sequences. Given an m-part assembly, i.e. an assembly with m parts, the parts are encoded by $1, \dots, m$. Assembly steps are also denoted by the set of integers $\{1, \dots, m\}$. An assembly sequence is denoted by a set of m pairs $\langle i, n \rangle$, where i is an assembly step and n denotes a part. By $\langle i, n \rangle$ we mean that part n is assembled at step i. So for any $\langle i, n \rangle$ and $\langle j, k \rangle$, $n \neq k$ if $i \neq j$.

If we use a unary predicate $p(X)$ to denote that X is a part of the assembly under consideration, the knowledge of parts can be represented by the following set of facts.

$$\{p(1), p(2), \dots, p(m)\}.$$

In order to represent an assembly sequence by ASP, a binary predicate $s(I, X)$ is introduced to denote that part X is assembled at step I. Then an assembly sequence for an m-part assembly can be represented by a set of m atoms $\{s(1, x_1), \dots, s(m, x_m)\}$, where $x_i \in \{1, \dots, m\}$ and $i \in \{1, \dots, m\}$.

Example: Given the assembly in Fig.1, if we use integer 1, 2, 3 and 4 to encode part a, b, c and d, respectively, an assembly sequence can be described as follows:

$$\{s(1,1), s(2,2), s(3,3), s(4,4)\}.$$

In this assembly sequence, the order for the parts to be assembled is 1-2-3-4.

For an m-part assembly, the constraints that any part of the assembly must be assembled can be described as:

For each part $k \in \{1, \dots, m\}$, there exists some step J such that $s(J, k)$ holds.

This constraint can be represented as m rules with each rule corresponding to one part:

$$s(1, 1) \vee \dots \vee s(m, 1). \dots s(1, m) \vee \dots \vee s(m, m).$$

The constraints that any part can be assembled only once can be described as:

*For each part X and two steps I and J,
if I!=J then s(I, X) and s(J, X) cannot be true simultaneously.*

In DLV language, the constraint can be described as follows:

$$\text{:- } s(I, X), s(J, X), I!=J.$$

When the constraint is instantiated, we will get m^3 instances.

3.2 Representation of the Assembly’s Connection Graph and Translation Function

ASP can represent an assembly’s connection graph in a straightforward manner. Let binary predicate $a(X,Y)$ denote that part X and part Y have a surface contact. The assembly’s connection graph in Fig.1(c) can be translated into ASP facts:

$$\{a(1,2).a(1,3).a(1,4).a(2,4).a(3,4).\}$$

and the following two rules:

$$\{a(X,Y):- a(Y,X). \quad \sim a(X,Y):- \text{not } a(X,Y), p(X), p(Y).\},,$$

where 1, 2, 3 and 4 are the encoding of part a, b, c and d, respectively.

The first rule says that the edges in a connection graph are undirected, and the second rule claims that the surface contact information is complete and hence can be used with closed world assumption, i.e. any contact information which can not be derived from the rules and facts in the above is false.

Translation function can be represented as ASP rules in a very natural manner. To do this we need to create a triple predicate $\text{pre}(X, D, Y)$, which means that part X does not have the freedom of translational motion w.r.t. part Y in direction D from far away, i.e. part Y prevent the motion of X in direction D. Therefore the predicate $\text{pre}()$ can be viewed as the *interference relation* between a pair of parts. By the definitions of $\text{pre}()$ and translation function T , the following proposition holds.

Proposition 3.1. Atom $\text{pre}(a, i, b)$ is true if and only if $T(b, i, a)=0$.

By the above proposition, the translation function of the assembly in Fig.1(a) can be translated into the following set of facts:

$$\{\text{pre}(2,1,1).\text{pre}(2,2,1).\text{pre}(2,5,1).\text{pre}(3,2,1).\text{pre}(4,1,1).\text{pre}(4,2,1).\text{pre}(4,3,1).\text{pre}(4,4,1).\text{pre}(4,6,1).\text{pre}(1,2,2).\text{pre}(1,4,2).\text{pre}(1,5,2).\text{pre}(3,2,2).\text{pre}(4,1,2).\text{pre}(4,2,2).\text{pre}(4,3,2).\text{pre}(4,4,2).\text{pre}(4,6,2).\text{pre}(1,5,3).\text{pre}(2,5,3).\text{pre}(4,1,3).\text{pre}(4,2,3).\text{pre}(4,3,3).\text{pre}(4,4,3).\text{pre}(4,6,3).\text{pre}(1,1,4).\text{pre}(1,3,4).\text{pre}(1,4,4).\text{pre}(1,5,4).\text{pre}(1,6,4).\text{pre}(2,1,4).\text{pre}(2,3,4).\text{pre}(2,4,4).\text{pre}(2,5,4).\text{pre}(2,6,4).\text{pre}(3,1,4).\text{pre}(3,3,4).\text{pre}(3,4,4).\text{pre}(3,5,4).\text{pre}(3,6,4).\}$$

3.3 Representation of Connectivity Constraints and Precedence Constraints

Straightforward Representation for Connectivity Constraints (SR-CC)

Given an assembly and a connection graph, the connectivity constraints require that the last assembled part must have surface contact with at least one part that has been assembled. If we use the predicate $a(X, Y)$ this constraints can be described in a very natural way. All the constraints are of the following form:

$$\text{:- } s(n, X_n), s(n-1, X_{n-1}), \dots, s(1, X_1), \sim a(X_n, X_{n-1}), \dots, \sim a(X_n, X_1). \tag{A.1}$$

The above constraint says that if the most recently assembled part is X_n , X_n cannot have surface contacts with none of the parts assembled in the previous steps. If an assembly has m parts, for each $n \in \{2, \dots, m\}$, there will be a constraint of the form (A.1); each constraint will have m^n ground instances. So in total there will be $\sum_{n=2}^m (m^n)$ instances. Therefore it is predictable that this representation of the connectivity constraints has an exponential space complexity. This motivates us to find more efficient representation, where each constraint contains a smaller number of variables.

Improved Representation for Connectivity Constraints (IR-CC)

Recall the image of a set of elements in a graph. Given an assembly's connection graph $\langle P, C \rangle$ and a set $G \subseteq P$ of nodes, the image of G in the graph is defined as:

$$\text{Image}(G) = \{e_2 | \langle e_1, e_2 \rangle \in C \wedge e_1 \in G\}.$$

If a part i is assembled in the first step and part j is assembled in the second step, i.e. $s(1, i)$ and $s(2, j)$ holds, then j must be an element of $\text{Image}(\{i\})$ in the connection graph. Generally if $s(n, j)$ hold, j must belong to the image of the set of parts assembled from step 1 to step $(n-1)$. In order to represent this knowledge, we introduce a set of constants $\{t_1, \dots, t_n\}$, where t_i denotes the image of the set of parts assembled from step 1 to step i . This denotation immediately leads to the following relation:

$$t_i \subseteq t_j, \text{ where } j \geq i > 0. \quad (\text{A.2})$$

Formally t_i can be defined recursively as follows:

Definition 3.2. Let X and Y be two parts of an assembly,

- 1) if $s(i, Y)$ and $a(X, Y)$, then $X \in t_i$;
- 2) if $X \in t_{i-1}$, then $X \in t_i$;
- 3) the elements of t_i are generated only by 1) and 2).

If we use $\text{bel}(X, t_i)$ to denote that X is an element of t_i , then $\text{bel}(X, t_i)$ can be defined as follows.

For $i=1$, we have:

$$\text{bel}(X, t_1) :- s(1, Y), a(X, Y). \quad (\text{A.3})$$

$$\sim \text{bel}(X, t_1) :- \text{not bel}(X, t_1), p(X). \quad (\text{A.4})$$

For $m \geq i \geq 2$, we have:

$$\text{bel}(X, t_i) :- s(i, Y), a(X, Y). \quad (\text{A.5})$$

$$\text{bel}(X, t_i) :- \text{bel}(X, t_{i-1}). \quad (\text{A.6})$$

$$\sim \text{bel}(X, t_i) :- \text{not bel}(X, t_i), p(X). \quad (\text{A.7})$$

For an m -part assembly, there will be m^2 ground instances of constraint (A.3), m ground instances of constraint (A.4), $m^2(m-1)$ ground instances of constraint (A.5), and $m(m-1)$ ground instances of constraint (A.6) or (A.7). So there are a total of

(m^3+3m^2-2m) ground constraints instances. With this definition the connectivity constraints can be described as follows:

*if $s(i, X)$ holds, then $bel(X, t_{i-1})$ must hold; or equivalently,
 $s(i, X)$ and $\sim bel(X, t_{i-1})$ cannot be true simultaneously.*

In answer set programming, the constraints can be described as:

$$:- s(i, X), \sim bel(X, t_{i-1}). \quad (m \geq i \geq 2) \quad (\text{A.8})$$

Given an m -part assembly, there will be $(m-1)$ constraints of form (A.8), and each constraint will have m instances. So in total there are $m(m-1)$ instances. Then in order to define connectivity constraints, we need $[m(m-1)+m^3+3m^2-2m]=(m^3+3m^2-3m)$ ground constraint instances, which is a great improvement compared with the straightforward method for representing the same constraints.

Straightforward Representation for Precedence Constraints (SR-PC)

The precedence constraints require that any part considered for assembling cannot be prevented in all six directions by the previously assembled parts. This constraint has clear relationship with the translation function and therefore the predicate $pre()$.

Now we give the most natural representation of this kind of constraint. For an m -part assembly, we first introduce a set of constants $\{a_1, \dots, a_m\}$, where a_i denotes the set of parts assembled before step i . Apparently there is no parts belonging to a_1 . And the membership of a part in a_i ($m \geq i \geq 2$) can be defined easily as follows.

For $i=2$, we have:

$$bel(X, a_2):- s(1, X). \quad (\text{B.1})$$

For $m \geq i > 2$, we have:

$$bel(X, a_i):- s(i-1, X). \quad (\text{B.2})$$

$$bel(X, a_i):- bel(X, a_{i-1}). \quad (\text{B.3})$$

Let's suppose that we have assembled $(i-1)$ parts of an assembly, then the constraint for assembling the i -th ($m \geq i \geq 2$) part can be described as follows:

$$\begin{aligned} :- s(i, X), pre(X, 1, X_1), pre(X, 2, X_2), pre(X, 3, X_3), pre(X, 4, X_4), \\ pre(X, 5, X_5), pre(X, 6, X_6), bel(X_1, a_i), bel(X_2, a_i), bel(X_3, a_i), \\ bel(X_4, a_i), bel(X_5, a_i), bel(X_6, a_i). \end{aligned} \quad (\text{B.4})$$

Constraint (B.4) says that (a) part X is assembled at step i , and (b) X cannot be assembled at step i , i.e. in all six directions the motion of X is prevented by some previously assembled part, cannot be true simultaneously.

For an m -part assembly there will be m instances of (B.1), $m(m-2)$ instances of (B.2) or (B.3), and $m^7(m-1)$ instances of (B.4). A huge number!

Improved Representation for Precedence Constraints (IR-PC)

The space complexity of the SR-PC motivates us to find more efficient representation method, where there are fewer variables present in each constraint. In doing so we introduce the predicate $prevent(a_i, d, X)$ denoting that there is a part in

a_i , which prevents part X in direction d . Obviously $\text{prevent}(a_i, D, X)$ can be defined as follows:

$$\text{prevent}(a_i, D, X):- \text{pre}(X, D, Y), \text{bel}(Y, a_i). \quad (m \geq i \geq 2, D \in \{1, \dots, 6\}) \quad (\text{B.5})$$

With this predicate the constraint (B.4) can be rewritten as:

$$\begin{aligned} & :- \text{s}(i, X), \text{prevent}(a_i, 1, X), \text{prevent}(a_i, 2, X), \text{prevent}(a_i, 3, X), \\ & \quad \text{prevent}(a_i, 4, X), \text{prevent}(a_i, 5, X), \text{prevent}(a_i, 6, X). \end{aligned} \quad (\text{B.6})$$

The number of ground instances of constraints (B.5) is $6m^2(m-1)$; and the number of instances of constraint (B.6) is $m(m-1)$. So in total we have $m+2m(m-2)+6m^2(m-1)+m(m-1)=(6m^3-3m^2-4m)$ ground instances for constraints (B.1), (B.2), (B.3), (B.5) and (B.6). This is a great improvement over the straightforward formulation, where there are $(m^8-m^7+2m^2-3m)$ constraint instances.

3.4 ASP Programs for Assembly Sequences Generation

Given an assembly and its corresponding connection graph and translation function, the rules and constraints discussed in the above sections can be created automatically. Those rules and constraints will constitute an ASP program for generating all feasible assembly sequences for an assembly. Each program is divided into EDB consisting of an assembly's contact and interference information and IDB consisting of rules and constraints a feasible assembly sequence must satisfy. Note that the rules and constraints in an IDB are general in that they are shared by all assemblies with the same number of parts; and EDB, however, is the component that is exposed to frequent changes in the product design process. In the following sections, an ASP program in the language of DLV/smodels/cmodels is called a DLV/smodels/emodels program.

Section 3.3 has presented straightforward and improved representations for connectivity constraints and precedence constraints. Each combination of the representation methods for the two types of constraints leads to an ASP based method for MASP. So there will be four MASP methods that use: 1) SR-CC and SR-PC, 2) SR-CC and IR-PC, 3) IR-CC and SR-PC, and 4) IR-CC and IR-PC. Here we are mainly concerned with the first and fourth methods. In what follows the two methods will be denoted by Straightforward Method (SM) and Improved Method (IM), respectively. The space complexity of knowledge representation in SM and IM is presented in table 1.

Table 1. Space complexity of the knowledge representation in SM and IM, where m is the number of parts of the assembly

MASP Method	Representation method	Needed constraints	Total Number of ground instances	Space complexity
SM	SR-CC	(A.1)	$\sum_{n=2}^m (m^n)$	$O(m^m)$
	SR-PC	(B.1) ~ (B.4)	$m^8-m^7+2m^2-3m$	
IM	IR-CC	(A.3) ~ (A.8)	m^3+3m^2-2m	$O(m^3)$
	IR-PC	(B.1) ~ (B.3), (B.5), (B.6)	$6m^3-3m^2-4m$	

From table 1, it can be seen that a DLV program that adopts SM has an exponential space complexity, while a DLV program that adopts IM has polynomial space complexity $O(m^3)$. In section 4 we will further investigate the performance of the two methods on a collection of assemblies.

Take the four-part assembly in Fig.1 as an example, the DLV program that uses IM for constraints representation is shown in Fig.2.

```

% EDB consisting of the assembly's connection and interference information.
a(1,2).a(1,3).a(1,4).a(2,4).a(3,4).
% the assembly's interference relation.
pre(2,1,1).pre(2,2,1).pre(2,5,1).pre(3,2,1).pre(4,1,1).pre(4,2,1).pre(4,3,1).pre(4,4,1).pre(4,6,1).pre(1,2,2).pre(1,4,2).
pre(1,5,2).pre(3,2,2).pre(4,1,2).pre(4,2,2).pre(4,3,2).pre(4,4,2).pre(4,6,2).pre(1,5,3).pre(2,5,3).pre(4,1,3).pre(4,2,3).
pre(4,3,3).pre(4,4,3).pre(4,6,3).pre(1,1,4).pre(1,3,4).pre(1,4,4).pre(1,5,4).pre(1,6,4).pre(2,1,4).pre(2,3,4).pre(2,4,4).
pre(2,5,4).pre(2,6,4).pre(3,1,4).pre(3,3,4).pre(3,4,4).pre(3,5,4).pre(3,6,4).
% IDB consisting of constraints a feasible assembly sequence must satisfy.
p(1). p(2). p(3).p(4).      a(X, Y):- a(Y, X).
% the constraints that any part must be assembled at least once.
s(1,1) v s(1,2) v s(1,3) v s(1,4).      s(2,1) v s(2,2) v s(2,3) v s(2,4).
s(3,1) v s(3,2) v s(3,3) v s(3,4).      s(4,1) v s(4,2) v s(4,3) v s(4,4).
% the constraints that any part can be assembled only once.
:- s(Y, X), s(Z, X), Y!=Z.
% connectivity constraints.
:-s(2,X), ~bel(X,t1).      :-s(3,X), ~bel(X,t2).      :-s(4,X), ~bel(X,t3).
bel(X,t1):- s(1,Y), a(X,Y). ~bel(X,t1):- not bel(X,t1), p(X).
bel(X,t2):- s(2,Y), a(X,Y). bel(X,t2):- bel(X,t1).      ~bel(X,t2):- not bel(X,t2), p(X).
bel(X,t3):- s(3,Y), a(X,Y). bel(X,t3):- bel(X,t2).      ~bel(X,t3):- not bel(X,t3), p(X).
% precedence constraints.
:- s(2,X), prevent(a2,1,X), prevent(a2,2,X), prevent(a2,3,X),prevent(a2,4,X),prevent(a2,5,X),prevent(a2,6,X).
:- s(3,X), prevent(a3,1,X), prevent(a3,2,X), prevent(a3,3,X),prevent(a3,4,X),prevent(a3,5,X),prevent(a3,6,X).
:- s(4,X), prevent(a4,1,X), prevent(a4,2,X), prevent(a4,3,X),prevent(a4,4,X),prevent(a4,5,X),prevent(a4,6,X).
prevent(a2,D,X):- pre(X,D,Y), bel(Y,a2).      prevent(a3,D,X):- pre(X,D,Y), bel(Y,a3).
prevent(a4,D,X):- pre(X,D,Y), bel(Y,a4).      bel(X,a2):- s(1,X), bel(X,a3):- s(2,X), bel(X,a3):- bel(X,a2).
bel(X,a4):- s(3,X).      bel(X,a4):- bel(X,a3).

```

Fig. 2. A DLV program for generating assembly sequences for the assembly in Fig.1 with IM

Take the program in Fig.2 as input, DLV calculates 4 answer sets for the program.

$$\begin{array}{ll}
 \{s(1,1), s(2,2), s(3,3), s(4,4)\} & \{s(1,1), s(2,3), s(3,2), s(4,4)\} \\
 \{s(1,2), s(2,1), s(3,3), s(4,4)\} & \{s(1,3), s(2,1), s(3,2), s(4,4)\}
 \end{array}$$

Each answer set listed above corresponds to a feasible assembly sequence.

If we want to change the design of the assembly by altering the size of some parts, the contact and interference information may be changed. In this case we only need to substitute the EDB with a new EDB and reuse the original IDB.

For example, if the connection information of the EDB is changed to

$$\{a(1,2).a(1,3).a(1,4).a(3,4).\}$$

we only have to run the same IDB in Fig.2 on this new EDB to find a feasible assembly sequence.

4 Experiments

We have written C programs which, given a connection graph and a translation function for an assembly, automatically generate the ASP program for generating all the

assembly sequences for the assembly. Two experiments are performed. Firstly, SM and IM are implemented in DLV language, whose performance are tested on several assemblies with various number of parts. The results are shown in the first four rows of table.1; Secondly, we compare the performance of IM implemented in the languages of DLV, smodels, and cmodels, respectively. The results are shown in the remaining rows of table.2. In both experiments, we are interested in the time and memory efficiency of the two methods. The computer we use has a Pentium(R)4 CPU 3.00GHz and 1.0 GB memory. The situations for memory requirements for assemblies are obtained by monitoring the Windows Task Manager and the main concern is whether the memory resources run out on an assembly.

Table 2. Performance of SM and IM, where the row time in column m denotes the time in seconds for generating one assembly sequence for the m -part assembly chosen for our experiment, and *time-out* indicates that the runtime exceeds a limit of 5 hours

Number of parts in the assembly			4	6	8	11	14	15	16	20
SM	DLV	<i>time(s)</i>	0.59	1.03	202.2	time-out	time-out	time-out	time-out	time-out
		<i>memory out</i>	no	no	no	yes	yes	yes	yes	yes
IM	DLV	<i>time(s)</i>	0.02	0.02	0.05	0.48	3.81	3.35	0.39	time-out
		<i>memory out</i>	no	no	no	no	no	no	no	no
	smodels	<i>time(s)</i>	0.05	0.09	0.19	0.78	129.57	83.56	30.77	time-out
		<i>memory out</i>	no	no	no	no	no	no	no	no
	cmodels	<i>time(s)</i>	0.04	0.09	0.20	0.28	3.21	0.62	4.84	6.37
		<i>memory out</i>	no	no	no	no	no	no	no	no

The results of first experiment shows that the SM implemented in DLV language leads to system memory out quickly, and the IM, however, can produce a feasible assembly sequence in a short time on the assemblies with no larger than 16 parts.

The second experiment shows that DLV outperforms smodels on all experimental assemblies; and cmodels is more efficient than DLV on assemblies with more than 10 parts, with only one exception of the 16-part assembly. Typically, when the number of parts of an assembly reaches 20, both DLV and smodels cannot produce a feasible sequence in 5 hours; cmodels, however, gives a solution in a fairly short time (6.37 seconds). This performance is acceptable for MASP since it is not time critical.

Since the space complexity of IM is polynomial, the above results suggest two ways to apply IM to larger scale MASP problems. Using a fast answer set solver such as cmodels is the most obvious candidate, but this method has limited applications since it can be seen that the time complexity of IM using cmodels is not linear. The other way is to make use of the subassembly identification techniques to divide an assembly into several subassemblies, each of which has a small number of parts and therefore the feasible assembly sequences for which can be found quickly. The overall assembly sequence can be obtained by concatenating the assembly sequences of each individual subassembly. The second way is very promising and has the potential to make ASP based MASP method competitive for industrial production processes.

5 Conclusion and Discussion

This paper proposes an ASP based method for solving the NP-complete assembly sequence planning problem, with the aim to improve the information reuse in this process. The division of an ASP program into EDB and IDB provides a natural scheme

for this purpose. It is shown that once the IDB component is created for an assembly, it is reused/shared by all assemblies with the same number of parts. Compared with other approaches for assembly sequence planning, this is a great advantage.

Experiments are conducted to test the performance of our methods by using different answer set solvers. It is shown that *cmodels* out-performs *DLV* and *smodels* on most non-trivial assemblies, and is a very promising tool for solving MASP problems. In the future, we will integrate our method with existing CAD systems, which will make it possible to measure the information reuse brought by our method in the product development process.

References

1. Baldwin, D.F., Abeel, T.E., et al.: An integrated computer aid for generating and evaluating assembly sequences for mechanical products. *IEEE Trans. Auto. Con.* 7(1), 78–94 (1991)
2. Baral, C., Gelfond, M.: Logic programming and knowledge representation. *Journal of Logic Programming* 19, 73–148 (1994)
3. Gelfond, M., Lifschitz, V.: Classic negation in logic programs and disjunctive databases. *Next Generation Computing* 9, 365–385 (1991)
4. Giunchiglia, E., Lierler, Y., Maratea, M.: *Cmodels-2: SAT-Based Answer Set Programming*. In: *Proceedings of AAAI* (2004)
5. Gottipolu, R.B., Ghosh, K.: A simplified and efficient representation for evaluation and selection of assembly sequences. *Computer in Industry* 50(3), 251–264 (2003)
6. Gu, T., Liu, H.: The symbolic OBDD scheme for generating mechanical assembly sequences. *Formal Methods in System Design* 33(1–3), 29–44 (2008)
7. de Mello, L.S.H., Sanderson, A.C.: A correct and complete algorithm for the generation of mechanical assembly sequences. *IEEE Trans. Auto. Con.* 7(2), 228–240 (1991)
8. de Mello, L.S.H., Sanderson, A.C.: Representation of mechanical assembly sequences. *IEEE Transactions on Robotics Automation* 7(2), 211–227 (1991)
9. Hong, D.S., Cho, H.S.: Generation of robotic assembly sequences using a simulated annealing. In: *Proceedings of the IEEE/RSJ IROS 1999*, pp. 1247–1252 (1999)
10. Leone, N., Pfeifer, G., et al.: The *DLV* system for knowledge representation and reasoning. *ACM TOCL* 7(3), 499–562 (2006)
11. Lifschitz, V.: Answer set programming and plan generation. *Artificial Intelligence* 138(1–2), 39–54 (2002)
12. Marian, R.M., Luong, L.H.S., et al.: Assembly sequence planning and optimization using genetic algorithms. *Applied Soft Computing* 2(3), 223–253 (2003)
13. Niemela, I., Simons, P., Syrjanen, T.: *Smodels: A System for Answer Set Programming*. In: *Proceedings of NMR 2000*, April 9–11 (2000)
14. Reiter, R.: On closed world datahases. In: Gallaire, Minker (eds.) *Logic and Datahases*, pp. 55–76. Plenum Press, New York (1978)
15. Romney, B., Godard, C., et al.: An efficient system for geometric assembly sequence generation and evaluation. In: *Proceedings of the 1995 ASME International Computers in Engineering Conference*, pp. 699–712 (1995)
16. Su, Q.: Applying case-based reasoning in assembly sequence planning. *International Journal of Product Research* 45(1), 29–47 (2007)
17. Wilson, R.H.: Minimizing user queries in interactive assembly planning. *IEEE Transactions on Automatic Control* 11(2), 308–311 (1995)
18. Yin, Z.P., Han, D., et al.: A connector-based hierarchical approach to assembly sequence planning for mechanical assemblies. *Computer Aided Design* 35(1), 37–56 (2003)

Manifold Alpha-Integration

Heeyoul Choi¹, Seungjin Choi^{2,3}, Anup Kataké⁴,
Yoonseop Kang², and Yoonsuck Choe¹

¹ Department of Computer Science and Engineering
Texas A&M University, USA
{hchoi,choe}@cs.tamu.edu

² Department of Computer Science

³ Division of IT Convergence Engineering
POSTECH, Korea

{seungjin,e0en}@postech.ac.kr

⁴ Starvision Technologies, Inc., Texas, USA
akatake@starvisiontech.com

Abstract. Manifold learning has been successfully used for finding dominant factors (low-dimensional manifold) in a high-dimensional data set. However, most existing manifold learning algorithms only consider one manifold based on one dissimilarity matrix. For utilizing multiple manifolds, a key question is how different pieces of information can be integrated when multiple measurements are available. Amari proposed α -integration for stochastic model integration, which is a generalized averaging method that includes as a special case arithmetic, geometric, and harmonic averages. In this paper, we propose a new generalized manifold integration algorithm equipped with α -integration, *manifold α -integration* (MAI). Interestingly, MAI can be shown to be a generalization of other integration methods (that may or may not use manifolds) like kernel fusion or mixture of random walk. Our experimental results also confirm that integration of multiple sources of information on individual manifolds is superior to the use of individual manifolds separately, in tasks including classification and sensorimotor integration.

1 Introduction

In data analysis, it is important to understand the structure of the data, which can be described as a manifold. Manifold learning involves inducing a smooth nonlinear low-dimensional manifold from a set of data points drawn from the manifold that is embedded in a high-dimensional space. Various manifold learning methods have been developed and have drawn much attention in pattern recognition and signal processing [1]. However, most existing manifold learning algorithms only consider one manifold based on one dissimilarity (or distance) matrix. Since different measurements generate data sets on different manifolds, the resulting manifold needs to be integrated into one to use all the structural information from different measurements in the framework of manifold learning.

How can different measurements given by different distance matrices be used together to form an integrated manifold? A key question here becomes how different pieces of information can be integrated. In pattern recognition systems, data integration has been an important issue to improve accuracy relative to a single source of information because one sensor might not be good enough to provide unambiguous information. Some algorithms have been applied to integrate multiple sources of information (see [2] and references therein). However, each integration algorithm works optimally only with specific types of data sets. A more general approach, α -integration, was proposed by [3] for stochastic model integration of multiple positive measures. It is a one-parameter family of integration, where the single parameter α determines the characteristics of integration. Given a number of stochastic models in the form of probability distributions, it finds the optimal integration of the sources in the sense of minimizing α -divergence [3].

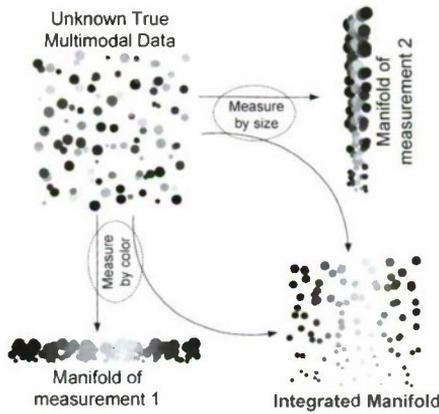


Fig. 1. An example of manifold integration when two manifolds are available from one data set. The two manifolds are from different measurements (color or size), and, each taken separately, is not suitable for understanding the data set perfectly. However, we can integrate the two measurements to obtain one integrated manifold which gives a complete picture of the data set.

Motivated by advances in manifold learning and data integration, in this paper, we propose a new manifold integration algorithm, *manifold α -integration* (MAI) that combines the manifold learning and data integration approaches as in Fig. 1. We show that our method includes as its special case previous methods such as the use of statistical distance [4], kernel-based data fusion [5] or mixture of random walks [6], by analyzing the compromised distances on the integrated manifold. Our experimental results with four data sets including real world data sets show promising results. Notably, we show that MAI can be applied to sensorimotor integration (cf. [7]).

2 Review of α -Integration

First, we provide a brief overview of α -integration, more details on which can be found in [3]. One exemplary application of α -integration can be found in [8] where α -integration successfully generalizes evidence theory. Let us consider two positive measures of random variable x , denoted by $m_1(x) > 0$ and $m_2(x) > 0$ for $i = 1, 2$. α -mean [3] is a one-parameter family of means, defined by

$$\tilde{m}_\alpha(x) = f_\alpha^{-1} \left(\frac{1}{2} \{ f_\alpha(m_1(x)) + f_\alpha(m_2(x)) \} \right), \tag{1}$$

where $f_\alpha(\cdot)$ is a differentiable monotonic function given by

$$f_\alpha(z) = \begin{cases} z^{\frac{1-\alpha}{2}}, & \alpha \neq 1, \\ \log z, & \alpha = 1. \end{cases} \tag{2}$$

The function $f_\alpha(\cdot)$ in Eq. (2) is the only function that enables α -mean to be linear scale free for $c > 0$, i.e., α -mean of $cm_1(x)$ and $cm_2(x)$ is $c\tilde{m}_\alpha(x)$, since

$$c\tilde{m}_\alpha(x) = f_\alpha^{-1} \left(\frac{1}{2} \{ f_\alpha(cm_1(x)) + f_\alpha(cm_2(x)) \} \right). \tag{3}$$

α -mean includes various commonly used means as its special case: for $\alpha = -1, 1, 3, \infty$ or $-\infty$, α -mean becomes arithmetic mean, geometric mean, harmonic mean, minimum, or maximum, respectively. The value of the parameter α (which is usually specified in advance and fixed) reflects the characteristics of the integration. As α increases, α -mean resorts more to the smaller of $m_1(x)$ or $m_2(x)$, while as α decreases, the larger of the two is considered with more weight [3].

α -mean can be generalized to the weighted α -mixture of M positive measures $m_1(x), \dots, m_M(x)$ with weights $\mathbf{w} = [w_1, w_2, \dots, w_M]$, which is referred to as α -integration of $m_1(x), \dots, m_M(x)$ with weights \mathbf{w} [3].

Definition 1 (α -integration). The α -integration of $m_i(x)$, $i = 1, \dots, M$, with weights w_i is defined by

$$\tilde{m}(x) = f_\alpha^{-1} \left(\sum_{i=1}^M w_i f_\alpha(m_i(x)) \right), \tag{4}$$

where $w_i > 0$ for $i = 1, \dots, M$ and $\sum_{i=1}^M w_i = 1$.

Given M positive measures, $m_i(x)$, $i = 1, \dots, M$, the goal of integration is to seek their weighted average $\tilde{m}(x)$ that is as close to each of the measures as possible, while how close two positive measures are is evaluated using a divergence measure. It was shown by [3] that α -integration $\tilde{m}(x)$ is optimal in the sense that the risk function

$$\mathcal{J}_\alpha[\tilde{m}(x)] = \sum_{i=1}^M w_i D_\alpha[m_i(x) \parallel \tilde{m}(x)] \tag{5}$$

is minimized, where $D_\alpha[m_i(x) \parallel \tilde{m}(x)]$ is the α -divergence of $\tilde{m}(x)$ from the measures $m_i(x)$ [3].

3 Manifold α -Integration

In this section, we propose a new manifold integration method using α -integration, which leads to *manifold α -integration* (MAI), and we show that it includes previous integration methods as a special case.

3.1 Algorithm: MAI

Let \mathbf{G} be a weighted graph with N nodes, representing a manifold. Then, the distance between two nodes on the k th manifold, $D_{ij}^{(k)}$, can be transformed into probability $P_{ij}^{(k)}$, the transition probability from the i th node to the j th node on the k th manifold. We simply use the Gaussian kernel which is given by

$$P_{ij}^{(k)} = \frac{1}{Z_i^{(k)}} e^{-\frac{D_{ij}^{(k)2}}{\sigma^{(k)2}}}, \tag{6}$$

where $Z_i^{(k)}$ is a normalization term so that the sum of transition probabilities from the i th node to all other nodes on the k th manifold is 1, and $\sigma^{(k)}$ is a parameter representing the standard deviation. Given C dissimilarity matrices, $\mathbf{D}^{(1)}, \mathbf{D}^{(2)}, \dots, \mathbf{D}^{(C)}$, we can get C probability matrices, $\mathbf{P}^{(1)}, \mathbf{P}^{(2)}, \dots, \mathbf{P}^{(C)}$.

There are two approaches in using α -integration on multiple manifolds: (1) using transition probability matrices and (2) using distance matrices. First, the transition probability from the i th node to the j th node on the k th manifold is given by Eq. (6). So, given $\mathbf{P}^{(1)}, \mathbf{P}^{(2)}, \dots, \mathbf{P}^{(C)}$, with α -integration, the compromised probability is given by

$$P_{\alpha,ij} = \frac{1}{Z_{\alpha,i}} f_{\alpha}^{-1} \left(\sum_{k=1}^C w_k f_{\alpha}(P_{ij}^{(k)}) \right), \tag{7}$$

where $Z_{\alpha,i}$ is a normalization term. From the compromised probability \mathbf{P}_{α} , we can reconstruct the compromised dissimilarity $\mathbf{D}_{\alpha p}$ as follows.

$$D_{\alpha p,ij} = \sigma^* \sqrt{-\log(P_{\alpha,ij})}, \tag{8}$$

where σ^* is the average of $\sigma^{(k)}$, $k = 1, \dots, C$.

Then we use kernel Isomap [9]. Given a distance matrix, $\mathbf{D}_{\alpha p}$, we substitute $\tilde{\mathbf{D}}_{\alpha p}$ for $\mathbf{D}_{\alpha p}$, which is given by

$$\tilde{D}_{\alpha p,ij} = D_{\alpha p,ij} + c(1 - \delta_{ij}), \tag{9}$$

where δ_{ij} is the Kronecker delta. Here, c is the solution of constant-shifting method [10] to make the doubly centered kernel matrix $\tilde{\mathbf{K}} = -\frac{1}{2} \mathbf{H} \tilde{\mathbf{D}}_{\alpha p}^2 \mathbf{H}$ positive semi-definite. Here, $\tilde{\mathbf{D}}_{\alpha p}^2$ is the element-wise square of $\tilde{\mathbf{D}}_{\alpha p}$ and $\mathbf{H} =$

$I - \frac{1}{N} \mathbf{e}_N \mathbf{e}_N^T$, where $\mathbf{e}_N = [1 \dots 1]^T \in \mathbb{R}^N$. Finally, after eigen-decomposition, $\widetilde{\mathbf{K}} = \mathbf{V} \mathbf{A} \mathbf{V}^T$, projection mapping \mathbf{Y} is given by

$$\mathbf{Y} = \mathbf{V} \mathbf{A}^{\frac{1}{2}}. \tag{10}$$

A more interesting approach is to apply α -integration to the distance matrices directly. Given C dissimilarity matrices, $\mathbf{D}^{(1)}, \mathbf{D}^{(2)}, \dots, \mathbf{D}^{(C)}$, we can reconstruct the compromised dissimilarity $\mathbf{D}_{\alpha d}$ directly by α -integration without considering the transition probability. It is given by

$$D_{\alpha d, ij} = f_{\alpha}^{-1} \left(\sum_{k=1}^C w_k f_{\alpha}(D_{ij}^{(k)}) \right). \tag{11}$$

Note that Eq. (7) is an α -integration of the probabilities, while Eq. (11) is an α -integration of the distances. So, with Eq. (7), the compromised distance in Eq. (8) is different from that of Eq. (11). We derive two slightly different manifold integration methods from these two different integration approaches, and call the two versions MAI_p and MAI_d, respectively. After getting \mathbf{D}_{α} (either $\mathbf{D}_{\alpha p}$ or $\mathbf{D}_{\alpha d}$), the rest is the same as kernel Isomap [9], by which our method inherits the dimensionality reduction property. Note again that since MAI uses kernel Isomap after obtaining the compromised distance matrix, it inherits the projection property of kernel Isomap which involves the projection of novel data points onto the associated low-dimensional space. Due to limited space, we do not derive the equations for the projection here. The derivations for the projection property can be found in [9].

3.2 Comparison with Existing Data Integration Approaches

We analyze MAI_p and MAI_d, comparing them to previous methods [4,5,6] even though some of them are not immediately about manifold integration.

Case 1: In *random walk on multiple manifolds* (RAMS) [4], the compromised transition probability matrix \mathbf{P}^* is simply given by multiplication of the source probabilities. Approximately, this is a special case of MAI_p in Eq. (7) with $\alpha = 1$ and uniform weights for all manifolds:

$$P_{1,ij} = \frac{1}{Z_{1',i}} f_1^{-1} \left(\sum_{k=1}^C \frac{1}{C} f_1(P_{ij}^{(k)}) \right) = \frac{1}{Z_{1',i}} (P_{ij}^*)^{\frac{1}{C}}, \tag{12}$$

where $Z_{1',i}$ is a normalization term. Then, the compromised distance in Eq. (8) is reconstructed by

$$D_{1p,ij} = \sigma^* \sqrt{\log Z_{1',i} - \frac{1}{C} \log P_{ij}^*}, \tag{13}$$

which is almost the same as the compromised distance in RAMS except for the normalization term and $\frac{1}{C}$. That is, RAMS is approximately a special case of

MAI_p with $\alpha = 1$ and uniform weights. If we relax the assumption on the weights so that the sum of weights is not 1 but $\sum_{k=1}^C w_k = C$, then MAI_p leads to an exactly the same result as RAMS.

Now, we can check the case when α -integration is applied to the distance matrices directly (MAI_d). When $\alpha = -3$ and the weights are given by $\frac{1}{\sigma^{(k)2}}$, the compromised distance matrix of MAI_d in Eq. (11) is given by

$$D_{-3d,ij} = \sqrt{\sum_{k=1}^C \frac{1}{\sigma^{(k)2}} (D_{ij}^{(k)})^2}, \tag{14}$$

which is the same as the the compromised distance matrix D^* in RAMS except the normalization terms. Here, the weights $\frac{1}{\sigma^{(k)2}}$ can serve as normalization terms for different units across measurements.

Case 2: [5] used a weighted sum of kernel matrices for kernel-based data fusion. For the special case when $\alpha = -3$ and weight w_k for the k th manifold applied to MAI_d, the corresponding kernel matrix K_{MAI_d} is given by just the weighted average of the kernel matrices as follows:

$$K_{MAI_d} = -\frac{1}{2}H(D_{-3d})^2H = \sum_{k=1}^C w_k K^{(k)}, \tag{15}$$

where $K^{(k)} = -\frac{1}{2}HD^{(k)2}H$. Notice that the last term in Eq. (15) is the kernel-based data fusion proposed in [5] which can now be seen as a special case of MAI_d. It was shown that manipulating the distance matrix gives a better result than manipulating the kernel matrix directly [9]. In other words, the integrated space of MAI_d can be better than (or at least equal to) the kernel-based data fusion methods when the α value is carefully chosen.

Case 3: Also, in [6], even though they did not discuss directly about manifold integration, a mixture of random walks was used as an integration method. With $\alpha = -1$ and different weights w_k for the k th manifold, MAI_p has

$$P_{-1,ij} = \frac{1}{Z_{-1,i}} \sum_k^C w_{ki} P_{ij}^{(k)}, \tag{16}$$

which is a mixture of random walks.

In sum, we checked three previous methods for data integration and compared them with our two proposed approaches. The previous approaches all turn out to be (approximately) a special case of our proposed method.

4 Experiments and Results

In order to show the effectiveness of our method, we carried out experiments with four different data sets: (1) disc data set made of 100 discs with different colors and sizes [4]; (2) head-related impulse response (HRIR) data [11]; (3) the CMU ARCTIC speech database [12]; and (4) sensorimotor integration.

4.1 Disc Data

We used an artificial disc data set to show the differences between the three methods: (1) RAMS, (2) MAI on transition probability matrices (MAI_p), and (3) MAI on distance matrices (MAI_d). Let $\mathbf{X} \in \mathbb{R}^{2 \times 100}$ be the discs' locations. The first row \mathbf{X}_1 and the second row \mathbf{X}_2 are the coordinates for color and size, respectively. From this disc data set, each distance matrix is obtained by only color or by size, respectively, and squared as follows.

$$D_{ij}^{(k)} = \text{dist}(X_{k,i}, X_{k,j})^2.$$

Fig. 2 shows the data set and three integrated spaces from the three methods. If we use $\alpha = 1$ and $\alpha = -3$ for MAI_p and MAI_d, respectively, then the results of MAI are the same as RAMS. Here, we chose the α values to be 0.89 and -1.25 for MAI_p and MAI_d, respectively. Note that the integrated space from MAI have almost a square shape, which is supposed to be like that, while RAMS has a fat square even though it found a "properly" integrated space where color and size are two dominant coordinates. In addition, MAI_d found almost the same result as the original set, while MAI_p has a little denser dots around the center of the space. Even though MAI_p generalizes RAMS with the same transition probability matrices, the reconstructed distance matrix is not optimal in the α -integration sense, because the transition probability equation in Eq. (6) is combined into f_α , which is not a linear scale free function of distance any more. This can be why MAI_p has a little distortion in the integrated space, even though it is still better than RAMS.

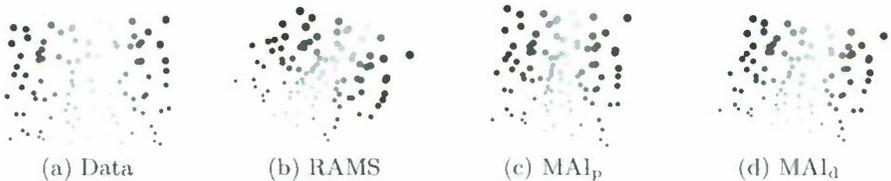


Fig. 2. Disc data set (a) and three integrated spaces (b-d); (a) original data set, 100 discs, (b) RAMS, (c) MAI_p with $\alpha = 0.89$, and (d) MAI_d with $\alpha = -1.25$

4.2 HRIR Data

In this experiment, we used the public-domain CIPIC HRTF data set [11] and applied kernel Isomap to each ear's HRIR data to generate a 2-dimensional manifold for each ear. Then we applied MAI_d to integrate the two manifolds. The detailed description for the HRIR data sets can be found in [11]. We mainly pay attention to the HRIRs involving sound sources specified by different elevation angles. The database contains HRIRs sampled at 1250 points around the head for 45 subjects. Azimuth is sampled from -80° to 80° and elevation from -45° to 230.625° . Each HRIR is a 200-dimensional vector corresponding to a duration of about 4.5ms.

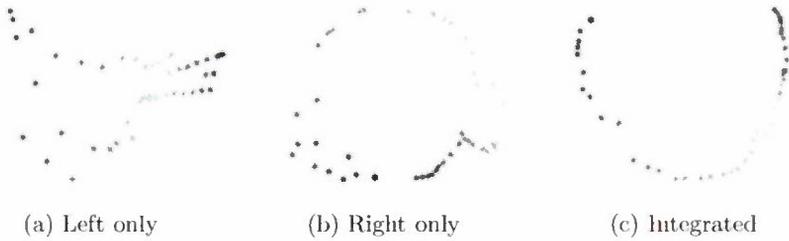


Fig. 3. Embedded manifolds of (a) left HRIR (b) right HRIR and (c) integrated HRIR. Even though the left HRIR is seriously distorted and the right one is also not smooth, the integrated space shows a very smooth, low error result, due to the use of both pieces of information.

Fig. 3 shows the performance of our method MAL_d with $\alpha = -0.5$ on 20th subjects in the data set. MAL_d was applied to the distance matrices of (a) and (b). Either (a) or (b) is not perfect for locating the sound source. The integrated result is better than the two results considered separately, as to where the sound source is. Note that the embedded manifolds in Fig. 3 have some ambiguities like up-down or front-back.

4.3 Speech Data

We carried out numerical experiments with the CMU ARCTIC speech database [12] in order to show an integrated manifold from multiple manifolds which leads to a speaker independent phoneme space and to show the benefit of the integrated space for phoneme classification. The CMU ARCTIC database was constructed as a phonetically balanced, US English single speaker database designed for unit selection speech synthesis research. The database includes US English male ('bdl', 'rms') and female ('slt', 'clb') speakers, each speaking a bunch of sentences. From the sentences, we extracted four phonemes, 'AH', 'EH', 'IH' and 'OW' for each speaker and converted each phoneme into Mel frequency cepstral coefficients (MFCCs), which served as our feature vectors.

Speaker independent phoneme space. First, we found one map of these vowels from four speakers' four vowels, where each phoneme consisted of 300 sample data points. Fig. 4 shows two speakers' individual maps from kernel Isomap. Even though they pronounced the same phonemes, their maps are different from each other. Furthermore, the clusters of phonemes are not well separated even in each map, since each map represents both linguistic information and speaker dependent information.

On the other hand, Fig. 5a is the integrated map from four speakers' maps, and it shows well defined clusters of phonemes, which means that this map represent the phoneme information but not speaker dependent information.

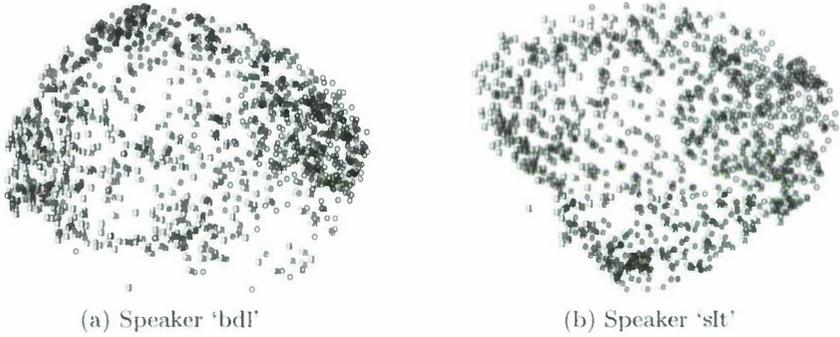


Fig. 4. Individual mapping of (a) speaker ‘bdl’ and (b) speaker ‘slt’ using kernel Isomap. Two maps look different because they are from different speakers even though they are for the same set of phonemes.

Classification. After getting the integrated map of phonemes, we tried to use this map for classification. We tested it with 6 different training data sizes. For each speaker’s individual phoneme, we randomly selected 50, 100, 150, 200, 250, and 300 samples for training and the rest for testing. For each trial, we repeated the experiment 30 times with randomly chosen data points and averaged them.

Fig. 5b shows the classification results with the quadratic classifier. From this figure, we can see that other speakers’ information is helpful for phoneme classification as long as the training data set is larger than a certain size. The average of classification rate for individual speaker data converges to 73.8% when 300 phonemes are used for training data, whereas MAI_d, especially with $\alpha = 3$,

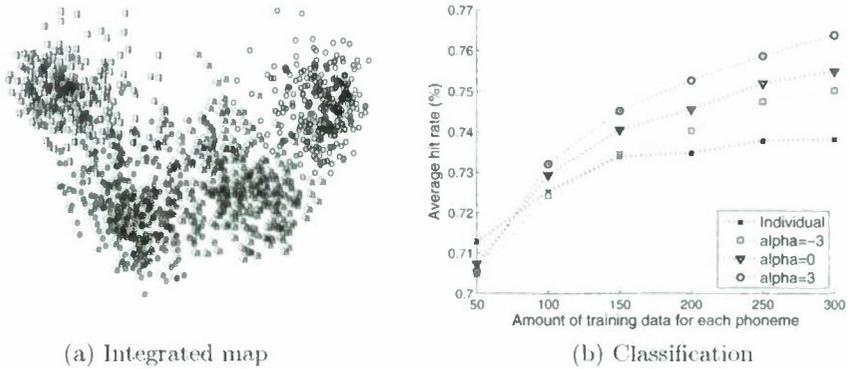


Fig. 5. Integrated manifold: (a) The map through MAI_d with $\alpha = 3$, where the phonemes are better clustered within class and separated from each other across class. (b) Hit rates for MAI_d (squares, triangles and circles) and individual map (crosses). After around 50 phonemes for training, MAI_d becomes better than the individual map. As shown above, RAMS and kernel-based data fusion are (approximately) special cases of MAI_d with $\alpha = -3$ (squares).

reaches 76.4%. Note that the performance changes as the α value changes and we can pick the best one to get better results. Here, $\alpha = 3$ is the best among integer values for α , which might be supported by maximum likelihood estimate (MLE). If we assume that each measurement has a Gaussian distribution which is almost the case for each phoneme in Fig. 5a, the MLE of the variance for all measurements is the harmonic mean of the individual variances. In this classification task, the best α value for the integration of distances might be explained as in the best α value for the variances. For more discussions about selecting the α values, see [13].

In Fig. 5b, however, when the training data size is smaller than around 80 phonemes, our proposed method is slightly worse than the individual-based map. The intersection is somewhere between 50 and 100. This phenomenon might be explained as follows. When the training data set is small, it is not enough to represent the real phoneme space. So, the test points could have been projected into a distorted map induced by the other speakers' information. But when the training data set is large enough, the projected space represents more likely the real phoneme space. So, from the test points, the speaker dependent noise is removed, which leads to better classification.

4.4 Sensorimotor Integration

To apply MAI to sensorimotor integration [7], we simulated sensory and motor information as shown in Fig. 6. The sensory information in (b), mimicking a non-linear transformation (e.g., log-polar transform) in the visual system, is a distorted version of the true square map in (a). The motor information in (e) is based on two angles of an articulated arm to reach locations within the true coordinate. The arm consisted of two sticks of same length. Given a point (t_x, t_y) in the true map with $t_x \in [0, 2]$ and $t_y \in [0, 2]$, the corresponding point in the sensory map (s_x, s_y) was given by

$$\begin{cases} s_x = t_x + 0.05, \\ s_y = \sqrt{t_y + 0.05}. \end{cases} \quad (17)$$

With the two sticks a_1 and a_2 pointing (t_x, t_y) with the end of a_2 , the corresponding point in the motor map (m_x, m_y) was calculated by

$$\begin{cases} m_x = \text{angle between } a_1 \text{ and the horizontal axis,} \\ m_y = \text{angle between } a_1 \text{ and } a_2, \end{cases} \quad (18)$$

where the angles are in radian.

In sensorimotor integration, two different information can be converted into a common representation (or integrated manifold in our words) for fusion, which is referred to as coordinate transformation [14]. [14] suggested maximizing mutual information between sensory information and motor information on the common representation while preserving topographic order to find two different mapping functions without considering structural information inherent in the data set,

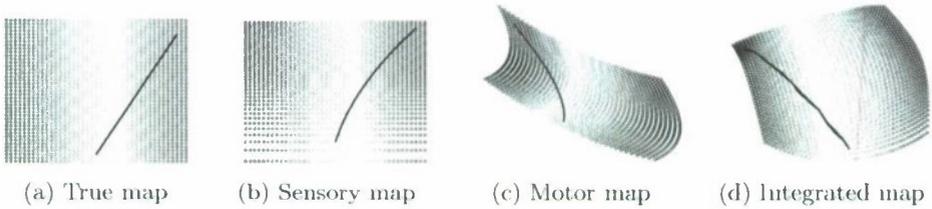


Fig. 6. Sensorimotor Integration: (a) true map and a straight line (reference), (b) sensory map and the projection of the reference line, (c) motor map and the projection of the reference line, and (d) an integrated map and the two projections of the reference line from the sensory and the motor space, respectively. We used MAI_d with $\alpha = 3$. The circles with the same color in the 4 maps represent the same location.

while we can simply obtain two such mapping rules from MAI, based on the structural information from the integrated manifold.

For example, when we draw a straight line on the true map, we get two kinds of information at the same time: sensory and motor, as the curves on the two maps (b) and (c). Though we cannot directly compare the two curves on the two different maps, with MAI we can project the two curves onto one integrated map and compare them as in (d). The blue curves are from the sensory space and the red curves from the motor space in (b), (c) and (d). In (d), the two curves closely overlap, though they are not perfectly the same because the maps are not perfect. This way, we can compare the sensory and the motor information directly on the integrated manifold which gives a common representation.

5 Conclusion

In this paper, we proposed a generalized manifold integration method utilizing α -integration which led to MAI. MAI integrates multiple measurements each of which is assumed to lie on a separate manifold. We showed that MAI includes as its special case the previous methods such as RAMS, kernel-based data fusion, or mixture of random walks. Furthermore, it can generalize to other integrated spaces in as many different ways as we want with a different α value. The experimental results confirmed that MAI integrates multiple measurements into one manifold in an effective manner, helping us to better understand the data set. For example, when we applied MAI to real world data sets, it found a better manifold than the individual manifolds. In classification tasks, the integrated manifold generally improved the accuracy when the training data set is reasonably large. Also, MAI was successfully applied to sensorimotor integration. The main contributions of this paper are as follows: (1) derivation of a generalized manifold integration algorithm and (2) showing that manifold integration is useful to many potential problems. We expect our results to serve as an effective framework for analyzing multimodal data sets on multiple manifolds. In this paper, we reconstructed the integrated space assuming that the α value is

manually chosen. In our future work, we intend to develop ways to find the α value automatically, optimized for the specific task.

Acknowledgments. This work was supported by Korea NRF Converging Research Center Program (No. 2009-0093714), NIPA ITRC support program (NIPA-2010-C1090-1031-0009), and NRF WCU Program (Project No. R31-2008-000-10100-0).

References

1. Seung, H.S., Lee, D.D.: The manifold ways of perception. *Science* 290, 2268–2269 (2000)
2. Hall, D.L., Llinas, J.: An introduction to multisensor data fusion. *Proceedings of the IEEE* 85(1), 369–376 (1997)
3. Amari, S.: Integration of stochastic models by minimizing α -divergence. *Neural Computation* 19, 2780–2796 (2007)
4. Choi, H., Choi, S., Choe, Y.: Manifold integration with Markov random walks. In: *Proc. Association for the Advancement of Artificial Intelligence (AAAI)*, Chicago, IL, vol. 1, pp. 424–429 (2008)
5. Lanckriet, G.R.G., Deng, M., Cristianini, N., Jordan, M.I., Noble, W.S.: Kernel-based data fusion and its application to protein function prediction in yeast. In: *Proc. Pacific Symposium on Biocomputing (PSB)*, Big Island, HI, vol. 9, pp. 300–311 (2004)
6. Zhou, D., Burges, C.: Spectral clustering and transductive learning with multiple views. In: *Proc. Int'l Conf. Machine Learning*, pp. 1159–1166 (2007)
7. Todorov, E.: Optimality principles in sensorimotor control. *Nature Neuroscience* 7(9), 907–915 (2004)
8. Choi, H., Katake, A., Choi, S., Choe, Y.: Alpha-integration of multiple evidence. In: *Proc. IEEE Int'l Conf. Acoustics, Speech, and Signal Processing*, Dallas, TX, pp. 2210–2213 (2010)
9. Choi, H., Choi, S.: Robust kernel Isomap. *Pattern Recognition* 40(3), 853–862 (2007)
10. Cailliez, F.: The analytical solution of the additive constant problem. *Psychometrika* 48(2), 305–308 (1983)
11. Algazi, V.R., Duda, R.O., Thompson, D.M., Avendano, C.: The CIPIC HRTF database. In: *Proc. 2001 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 99–102 (2001)
12. Kominek, J., Black, A.W.: CMU ARCTIC databases for speech synthesis (2003)
13. Choi, H., Choi, S., Katake, A., Choe, Y.: Learning alpha-integration with partially-labeled data. In: *Proc. IEEE Int'l Conf. Acoustics, Speech, and Signal Processing*, Dallas, TX, pp. 2058–2061 (2010)
14. Ghahramani, Z., Wolpert, D.M., Jordan, M.I.: Computational models of sensorimotor integration. *Science* 269, 1880–1882 (1997)

Ranking Entities Similar to an Entity for a Given Relationship

Yong-Jin Han¹, Seong-Bae Park¹, Sang-Jo Lee¹, Se Young Park^{1,*},
and Kweon Yang Kim²

¹ School of Electrical Engineering and Computer Science,
Kyungpook National University,
Daegu 702-701, Korea

{yjhan, sbpark, sjlee, sypark}@sejong.knu.ac.kr

² School of Computer Engineering, Kyungil University,
Gyeongsan 712-701, Korea
kykim@kiu.ac.kr

Abstract. This paper proposes a similarity ranking method for entities in the real world. Real world entities like people or objects often have some relationship between themselves. Finding such relationships from real world data can greatly enhance recognition of real world situations. However, it is difficult to capture such relationships from real world sensors alone. Nowadays, activities of people are often shared via Web. The activities can be represented as a relationship between people with shared items such as books, movies or other items. In semantic Web research, such relational information has been modeled in ontologies. The proposed ranking method of this paper is a method that finds meaningful relationships between entities in ontologies. In the first step, the method discovers pairs of entities which have meaningful connections in an ontology. Then it ranks the pairs according to similarities between entities. Unlike previous work, the proposed method assumes not only instance level connections, but also ontology schema level connections. This approach enables machines to access previously hidden indirect relationships into the similarity rankings. The experiments using an existing people-experience ontology show that the proposed method outperforms previous methods.

Keywords: Ranking entities, Ranking method, Semantic Association, Relationship.

1 Introduction

A machine can recognize people or other objects by using various machine learning methods and intelligent sensors. For example, an artificial neural network has been used to recognize people or objects by adapting its weight parameters with labeled voice data or labeled image data. However, the recognized entities are

* Corresponding author.

still fragmentary knowledge and they only represent the identity of the person or objects. In the real world, almost every entity is related with each other.

Comprehensive knowledge can be derived from such relationships between entities. To acquire deeper understanding of “contexts”, modeling and finding of such relationships are very important. For example, let’s assume that two people have been identified in the same room. And some smart sensors have recognized audio and video signals that could be understood as “task related with fire”. What do they actually do in the situation? To understand the situation, understanding of the context is essential. Also, expressed knowledge of previous relationship among entities (in this case, people and the fire) is also essential. If two people have a previous known relationship in the cooking session, and some knowledge about cooking and its relatedness with fire, can greatly enhance understanding of the context at hand.

It is not realistic to assume that such knowledge about entities, especially about people, can be captured by intelligent sensors networks. Nowadays, the activities of people are often shared via the web. Web 2.0 services like Twitter, Facebook, and Flickr are now recording various information about people in unstructured or semi-structured data. For example, people post their experiences about books or movies in blogs and microblogs, often with clear evidences like database links or semantic web tags.

With the advent of the semantic web, the activities have been modeled as an ontology which is in a machine-readable form [8]. Such an ontology provides sophisticated information about how people are related with each other. Thus, the common ground among people in the real world can be analyzed by using the existing ontologies that describe Web users’ activities.

In this paper, we are focused on analyzing the relationships among entities based on an ontology, when the entities (in this case, people) are recognized by a cognition system. Web ontology language (OWL) is a Semantic Web language designed to represent resources and publish them on the Web. OWL is a graph based representation of knowledge. In OWL, a unique entity is represented as an instance (a node) and the relationships between entities are represented as object properties (edges). Instances that have direct relationships have direct link between them. Also, it is possible to find indirect relationship by a third instance as a mediator. The former case is easy to find a relationship between instances, but the latter case can be discovered only through the paths composed of nodes (instances) and edges (properties).

Such a relationship can be interpretable by classes and properties of an ontology. Thus, in this paper, a relationship between instances belonging to a class is defined as a path of classes and properties. In order to discover more meaningful relationships, the relationship is restricted as a path which has a mediate class. For example, let an ontology have *Person* and *Food* as classes and *cook* as a property which links the two classes. Two instances, p_1 and p_2 are both instances of *Person*. They are both linked to an instance belonging to *Food*, with property of *cook*. Then, it is possible to declare that they have a relationship,

and the relationship in this case is $[p_1, \text{cook}, \text{Food}, \text{cook}, p_2]$. The two instances have identical path from each of them to the mediator, *Food*.

This paper proposes a method to find similar ontology instances, by ranking instances of the same class in terms of a given relationship. For example, if a person is given to the method with a target relationship, the proposed method will find similar people (instances of the same class) in terms of the given relationship. Thus, with the proposed method and a sufficient ontology, it is possible to find “Find related person with this person A, in terms of cooking”, or “List all personals that are similar to me, in terms of reading of philosophy books”.

There are several previous work on similar tasks [3,20,14]. Ranking or similarity methods of previous work are generally based on connected paths between entities. However, there could be some other meaningful relationships at the level of schema in an ontology, though two instances are not connected with any mediate instances. For example, if it is possible that two people do not share any instance directly (say, no same book), yet they share something in common in the level of schema (say, same genre of books, or same group of books).

To validate the approach, the relationship among people for a specific category of books and a genre of movies has been tested to find and rank similar people. The result have shown that the correlation between a labeled rank and a rank of the proposed method have positive correlation. The proposed method is also compared with two baseline methods. First method is a method that only compares paths that connect instances and disregards class level paths. The other method is a method that considers only the paths in the schema level. The proposed method outperforms both methods.

The rest of paper is organized as follows. Section 2 reviews some related works, and Section 3 presents how to discover entities with a relationship in an ontology. Section 4 explains ranking method in order to rank discovered entities by a given relationship. Section 5 shows the experimental results, and Section 6 concludes the paper.

2 Related Work

Kemafor et al. formalized relationships between instances for the RDF data model which is called semantic associations [11]. They defined four types of semantic associations for a given property sequence. For example, a simple semantic association between two instances is defined as a connected path from one of them to the other one through the property sequence. A property sequence explicitly expresses the meaning of a relationship. However, the property sequence can be interpreted in different way by classes where instances belong. Boanerges et al. [6] showed that such classes are useful to find a specific relationship. For example, when identifying money laundering, it is meaningful that a semantic association has an instance belonging to a class *Bank*. That is, relationships by a property sequence are discriminated by specifying the gaps between properties with classes.

In this paper, a relationship is defined without ambiguity by using a sequence of classes and properties. In addition, we focus on relationships among instances

belonging to the first class of a given class and property sequence. Thus what is important in this paper is not only to discover relationships from instances but also to measure similarities between relationships from one instance and relationships from the other one.

There are many previous work that tries to discover meaningful relationships between ontology instances [1,2,4]. Recently, SPARQL grammar is utilized for discovering semantic associations [12,13]. Not only the discovered associations are new information by itself, but also the path between instances can be used to measure the degree of relationships. In this paper, standard SPARQL is used to discover relationships as paths between instances.

A measurement for a relationship between classes were proposed in [16,17]. They used the probability information on instances related to a relationship between two classes. The work is similar to ours in the aspect that a relationship represented as classes and properties is measured. However, we are interest in relationships between instances while [16,17] only deals that of classes.

Boanerges et al. proposed six methods to measure the degree of relationships between instances [5]. They meaningfully considered a connected path between instances to measure relationship degree. Thus, the main difference between our work and others [16,17,5] is that not only connected path between instances but also connected paths through a schema are used to measure relationship degree between instances. Even though there is no connected path between two instances in instance level, some of paths from each of them are used to measure the relationship through the schema. More details are discussed in the next section.

Measuring relationships between instances is useful to various domains. Amit et al. [3] used it to find terrorist for national security and money laundering. They focused to discover paths between instances for a given property sequence. Recently, relationships between people entities expressed in ontologies were used to find social groups and social networks. Li et al. [6] proposed methods to integrate FOAF data and to extract social networks. Anna et al. [20] modeled networks of folksonomies and proposed a community dynamics notification algorithm to discover social networks from the network model. There are researches [18,9] that use ontologies as personal profiles to measure similarity between users respective preferences. The proposed method of this paper can enhance applications like preference modeling and social group finding, since the proposed method can reveal previously hidden indirect relationships.

3 Discovering Entities with a Common Relationship in an Ontology

A property sequence represents an implicit and complex relationship between instances of an ontology [11]. The meaning of such a relationship becomes clear by specifying classes [6]. In the viewpoint of [6], a relationship can be expressed as

$$P = [C_0, r_1, C_1, r_2, C_2, \dots, r_i, C_i, \dots, r_n, C_n],$$

where $C_i, 0 < i \leq n, n \geq 1$, are classes, $r_j, 1 \leq j \leq n$, are properties, C_i and C_{i+1} are linked by the property r_{i+1} in a given ontology.

That is, a relationship is a path in the schema of an ontology graph. In this paper, the first class C_0 is called a *source class* of a relationship P and the last one C_n is called a *target class* of P . Then a relationship P are discovered by finding a path between an instance of C_0 and an instance of C_n in the instance level.

There are many kinds of relationships as linked pairs of classes in an ontology. Above all relationships, this paper focuses on relationships among instances of a same class. Especially, it is meaningful that two instances have an identical kind of a relationship P . The target class of P is a mediator between two instances. Intuitively, it can be interpretable as that two entities may have common ground. Thus, in this paper, if two instances of a same class have an identical kind of a relationship, we say that *two instances have a common relationship*. A common relationship l_{xy}^P between two instances x and y for a given relationship P can be expressed as

$$l_{xy}^P = [x, C_1, r_1, \dots, r_n, C_n, r_n, \dots, r_1, C_1, y],$$

where $P = [C_0, r_1, C_1, r_2, C_2, \dots, r_i, C_i, \dots, r_n, C_n]$, x and y belong to C_0 , each x and y has a relationship, P . l_{xy}^P and l_{yx}^P are identical by the expression. Each instance, x and y is called a *source instance*. An instance of a target class is called a *target instance*. Note that both source instances need not to be connected with a target instance in order to have a common relationship.

In order to discover a common relationship for a given relationship P , source instances of P should be validated whether they have the relationship P . It is conducted by using a formal query language for an ontology and an existing reasoner. P is corresponding to a formal query to discover a relationship.

Final common relationships are decided as all of the pairs of positively validated instances. Thus, the number of instances with a common relationship is less than or equal to the number of 2-combinations of the source instances. Though the number of pairs of instances is finite, it increases exponentially with the number of source instances. However, all of common relationships can be discovered ahead of query time. In addition, our goal is replying such a query that "What is the most similar instance to an instance x for a given relationship P ". In this case, the number of candidate instances is equal to $(\# \text{ of source instances of } P) - 1$.

The focus of our current evaluations is involved measuring common ground among people based on an event ontology [19]. The ontology describes peoples' experiences for books or movies. Figure 1 shows examples of common relationships in the ontology.

The class *Person* is related with the class *Event* through the property *hasEvent*. *Event* is defined based on fundamental factors which describe an event. That is, an object of an event and a place and time related to the event is represented as properties of *Event*. The *Event* has two subclasses, *Appreciate* and

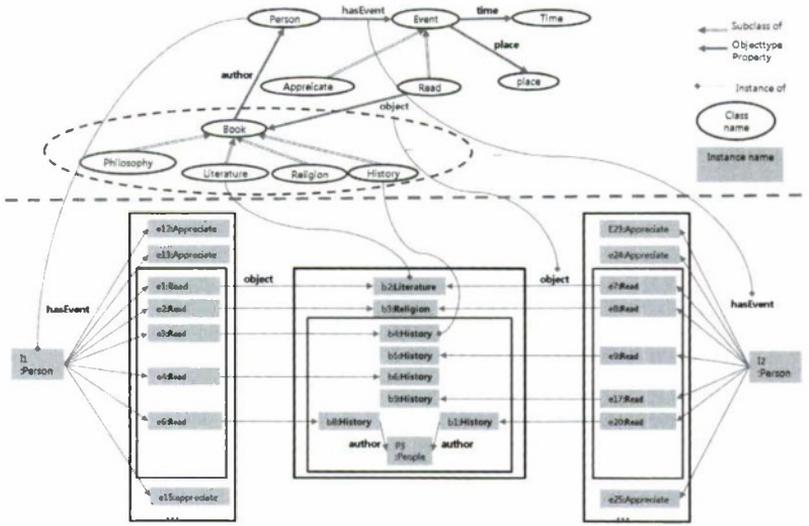


Fig. 1. Examples of common relationships in the event-ontology

Read. Read is related with Book through object. The Book has subclasses such as History, Literature, and Religion which are categories of books.

When a relationship is given as $P_1 = [Person, hasEvent, Read, object, Literature]$. The instances, I_1 of the class, Person has two paths which satisfy the relationship P_1 . Also, I_2 have such two paths. Thus, both of them have a common relationship for P_1 .

They are connected with two same target instances. However, a linked path in instance level is not a necessary condition for a common relationship. Let a relationship, P_2 be $[Person, hasEvent, Read, object, History]$. There is no target instance that connects the instances I_1 and I_2 for P_2 . However, two history books, b_1 and b_8 are a target instance from I_1 and that from I_2 respectively. Thus, they have a common relationship for P_2 . An interesting thing is that b_1 and b_8 are written by p_3 of Person. It means that the two people read a same author's books, though the books are different.

On the other hand, target instances, b_4 and b_6 of I_1 share no common property with target instances of I_2 . However, they have a connection with all of the target instances of I_2 in the schema level. That is, the target instances of I_1 are connected with them of I_2 through the mediator C_n . I_1 and I_2 also have a common relationship for P_2 in such a case. Though two people do not share any book, if each of them has been read many books of a same category, they may share something to talk with each other.

4 Ranking Common Relationships among Instances

We propose the following target function to rank common relationships for a relationship P .

$$f(l_{ab}^P, l_{cd}^P) = \begin{cases} 1 & \text{if } rel(l_{ab}^P) - rel(l_{cd}^P) > 0, \\ -1 & \text{if } rel(l_{ab}^P) - rel(l_{cd}^P) < 0, \end{cases} \text{ where } l_{xy}^P \in R_P.$$

R_P is a set of possible common relationships for P . The function, $rel(l_{xy}^P)$ is a measurement for the common relationship l_{xy}^P . It is a symmetric function on the two variables l_{ab}^P and l_{cd}^P . That is, $f(l_{ab}^P, l_{cd}^P)$ is equal to $-f(l_{cd}^P, l_{ab}^P)$.

The basic idea to measure the degree of relationships between two instances is that the more connections they have, the more similar they are. What is important is that the connections through a schema level should be considered for the measurement.

To do this, discovered relationships for P are grouped according to their source instances. Let G_x^P be a set of discovered relationships from a source instance x of a relationship P . Then, G_y^P is a set of relationships from y . The function $rel(l_{xy}^P)$ is realized by measuring similarity between these two groups.

Features of a group are represented as a vector to measure the similarity. They are corresponding to property values of target instances. A simple way to decide a feature value is to count the number of occurrences of property values from target instances. By doing this, connectivity by a target instance and their property values can be quantified by a similarity measure between two vectors.

An additional feature should be considered in order to measure connectivity between instances in the schema level. The following is a measurement for the connectivity from a source instance x to the target class C_n of a relationship P .

$$connectivity(x, C_n, P) = \frac{\# \text{ of paths in } G_x^P}{\# \text{ of paths from } x \text{ to instances in } C_1 \text{ through } r_1}$$

Each path of G_x^P starts with a hop from x to instances of C_1 through r_1 . However, not all paths starting with such hops belong to G_x^P . Thus the function *connectivity* quantifies how much an instance x allots for the target class C_n .

Therefore, without loss of generality, a feature vector of G_x^P can be expressed a weighted form. Let n is a number of distinct property values of target instances. Then a weighted vector for G_x^P is expressed as

$$v_x^P = [w_1 v_1^x, w_2 v_2^x, \dots, w_i v_i^x, \dots, w_n v_n^x, w_{n+1} v_{n+1}^x],$$

where $v_i^x, 1 \leq i \leq n$, are frequencies of property values from target instances, $v_{n+1}^x = connectivity(x, C_n, P)$, $w_j, 1 \leq j \leq n$, are weights for connectivity in instance level, w_{n+1} is a weight for connectivity in schema level, and $\sum_{i=1}^{n+1} w_i = 1$.

We are interest in the meaningfulness of the connectivity between instances in the schema level. For simplicity we assume that the weights in instance level are identical. Let $v_x^{P'}$ be a vector without the last feature of v_x^P . Then the function $rel(l_{xy}^P)$ can be defined by

$$rel(l_{xy}^P) = \left(\frac{\alpha}{n}\right)^2 v_x^{P'} \cdot v_y^{P'} + \beta^2 v_{n+1}^x v_{n+1}^y, \text{ where } \alpha = n w_1 \text{ and } \beta = w_{n+1}.$$

The instance-level connectivity is measured by the inner product between $v_x^{P'}$ and $v_y^{P'}$. Connectivity in schema level is measured by the product between v_{n+1}^x

and v_{n+1}^y . α is the total weight for connectivities in instance level. This function is equivalent to an inner product between v_x^P and v_y^P . The function rel is applied to the target ranking function f . Then the weights α and β can be adapted by using rank-labeled common relationships.

Note the ranking function provides general ranks for common relationships among instances. Ranks of instances similar to a given entity e is decided by sorting common relationships with e .

5 Experiment

To evaluate the proposed method, we use an event-ontology that describe people's blog postings for books, movies and IT-products [19]. We randomly selected 11 bloggers whose posting are more than thirty for each domain of books and movies. Six human annotators attended to label ranks of bloggers according to common ground with a blogger. Each annotator took up a blogger's position. Therefore an annotator ranked the other bloggers for a given query.

Table 1 is an example of bloggers' ranks for a query, "Rank the bloggers that are similar to your blogger in terms of reading of philosophy".

Table 1. Ranks of bloggers similar to a blogger in terms of reading of philosophy

	p_1	p_2	p_3	p_4	p_5	p_6
p_1	-	4	5	5	3	4
p_2	3	-	4	4	4	3
p_3	4	5	-	2	2	5
p_4	5	2	2	-	1	1
p_5	1	3	3	1	-	2
p_6	2	1	1	3	5	-
p_7	4	4	3	2	2	4
p_8	5	5	5	5	5	5
p_9	2	1	1	1	1	1
p_{10}	1	3	4	4	4	3
p_{11}	3	2	2	3	3	2

Annotators ranked 10 bloggers on positions of bloggers in the first row. The first column represents the ranked bloggers. For example, bloggers from p_2 to p_6 are sorted as p_5, p_6, p_2, p_3, p_4 on the position of p_1 . Ranks of the first five rows are used to adapt weight parameters of the proposed ranking method. The parameters were determined empirically by using correlations between labeled ranks and ranks by the proposed method. The method is tested with the last five row data.

The following 4 queries are used for the experiment.

- *Q1.* Rank the bloggers that are similar to your blogger in terms of reading of philosophy.
- *Q2.* Rank the bloggers that are similar to your blogger in terms of reading of economy.

- $Q3$. Rank the bloggers that are similar to your blogger in terms of appreciation of action.
- $Q4$. Rank the bloggers that are similar to your blogger in terms of appreciation of animation.

$Q1$ and $Q2$ are related with about categories of books. $Q3$ and $Q4$ are about genres of movies. These queries are corresponding to relationships in Table 2.

Table 2. Relationships corresponding to the test queries

Query	Relationship
$Q1$	[Person, hasEvent, Read, object, Philosophy]
$Q2$	[Person, hasEvent, Read, object, Economy]
$Q3$	[Person, hasEvent, Appreciate, object, Action]
$Q4$	[Person, hasEvent, Appreciate, object, Animation]

The proposed method discovers common relationships from the relationships in table 2. It used SPARQL [15] as a formal query language for an ontology, and jena API [7] as a reasoner.

Two base lines were tested for comparative analysis with the proposed method. First one f_i is a method that only compares paths connected by target instances. The method is simply modeled by setting the weight β of the proposed method to 0. The other one f_s is a method that considers only paths in the schema level. This method is given by setting the weight α to 0. The proposed ranking method is denoted as f_{i+s} .

Figure 2 shows experiment results of the three measurements for four queries. The x axis represents bloggers. For each of the bloggers, correlations between

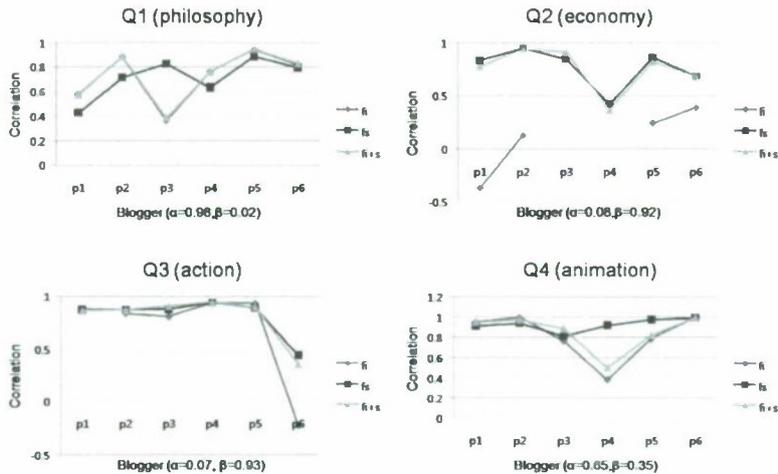


Fig. 2. The correlation between test data and the results by three measurements

labeled ranks and ranks by three measurements are recorded in y axis. Weights for the proposed method are marked under each graph. The bigger correlation means more similar to test data.

Both Q1 and Q2 ask the common ground for a book category. However, the two graphs shows very different results. f_i is relatively superior in Q1, but not in Q2. In addition, any target value of f_i is not given for p_3 and p_4 in Q2. It means that books read by p_3 and p_4 do not share any properties with others. Actually, their relationship group vectors are very sparse, since they read just two books of economy. Such sparseness is serious in Q2. p_2 and p_6 read the most number of books of economy and the number is just six. On the other hand, the six bloggers read at least eight books of philosophy. Therefore, it is possible to assume that if sufficient experiences are observed, f_i works well. A remarkable thing is that f_s meaningfully worked in Q2. The proposed method f_{i+s} reflects such a tendency that which one is more meaningful between f_i and f_s . That is, f_{i+s} are nearly identical with f_i in Q1 and f_s in Q2.

Both Q3 and Q4 are related to genre of movies. Most of the bloggers appreciated at least ten movies. Unusually, p_1 in Q3 appreciated just one movie of action. Thus a target value of f_i cannot be determined for p_1 . The most number of movies is 64 and 26 for each of action and animation. Thus, both the cases are relatively free for the sparseness problem than the cases of Q1 and Q2. f_i gives more than 0.7 correlation in most of the cases for Q3 and Q4. However, f_s gives more consistent results than f_i . It means that the participants are interested in unseen movies as well as already seen movies. The proposed method f_{i+s} shows an improved result of f_i by reflecting f_s .

Most of results by the proposed method f_{i+s} showed more than 0.6 of correlation. It outperformed than two baseline methods.

Then, how does the connectivity in schema level contribute to rank similar instances? In order to answer this question, schema-level connectivities of six bloggers are presented in table 3.

Table 3. Schema-level connectivities from six blogger instances to target classes

	p_1	p_2	p_3	p_4	p_5	p_6
<i>Philosophy</i>	0.27	0.26	0.16	0.35	0.43	0.33
<i>Economy</i>	0.10	0.14	0.04	0.04	0.11	0.14
<i>Action</i>	0.04	0.23	0.33	0.23	0.20	0.29
<i>Animation</i>	0.22	0.05	0.14	0.16	0.26	0.21

In this experiment, a connectivity from an instance (a blogger) to a target class (a category or a genre) is equal to the ratio of the number of seen items in specific categories to the total number of seen items in a domain by a blogger. Intuitively, it can be considered as the degree of interest for a book category or a movie genre. Most of the ratios for economy are less than any others. We observed f_i are very meaningful for Q1 (philosophy) , but not for Q2 (economy). Especially, p_3 in philosophy and p_4 in animation show a definite tendency that less interest a

blogger has, more dependent on others' experience the blogger become. In other words, if people are interested in a specific domain, then common experiences for some books or movies are important to construct common ground to debate. However p_6 in action and p_2 in animation shows results against this tendency. It means that inexperienced objects could be meaningful to construct common ground irrespective the interest.

6 Conclusion

In this paper, we proposed a similarity ranking method for entities with a common relationship. A common relationship between instances is formalized as a bi-directional link of classes and properties. If two instances have an identical path pattern from each of them to an instance, they have a common relationship by the formalism. Thus hidden indirect relationships between instances can be discovered as a connected path through the schema of an ontology. The proposed ranking method uses not only connected path in instance level, but also path through the schema of an ontology. The experiment results shows our method is more correlated with people intuition than a method just considered connected paths between instances.

The proposed method can conduct to rank common relationships among entities. That is, when A , B , C , and D are different entities, the method can decide relative degree of relationship between any two pairs of them. This can be applicable such a case that find the best partner among candidates for a project. In future work, such a task will be studied by using the proposed method.

Acknowledgement

This research was supported by the Conversing Research Center Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology (2009-0082262).

References

1. Alexaki, S., Christophides, V., Karvounarakis, G., Plexousakis, D., Tolle, K.: The ICS-FORTH RDFSuite: Managing Voluminous RDF Description Bases. In: 2nd International Workshop on the Semantic Web, Hong-Kong (2001)
2. Alkhateeb, F., Baget, J.F., Enzenat, J.: Complex path queries for RDF. In: Poster paper in International Semantic Web Conference 2005, Galway, Ireland (2005)
3. Amit, S., Boanerges, A., Budak, I., Chris, H., Cartic, R., Clemens, B., Yashodhan, W., David, A., Sena, F., Kemafor, A., Krysz, K.: Semantic Association Identification and Knowledge Discovery for National Security Applications. Journal of Database Management on Database Technology for Enhancing National Security 16(1) (2005)
4. Barton, S.: Designing Indexing Structure for Discovering Relationships in RDF Graphs. In: Proceedings of the Datedo 2004. Annual International Workshop on Databases, Texts, Specifications, and Objects, Galway, Ireland, pp. 7-17 (2004)

5. Boanerges, A., Christian, H., Budak, A., Cartic, R., Amit, P.: Ranking Complex Relationships on the Semantic Web. *IEEE Internet Computing* 9(4), 37–44 (2005)
6. Boanerges, A., Chris, H., Budak, A., Clemens, B., Amit, S.: Context-Aware Semantic Association Ranking. In: *The 1st International Workshop on Semantic Web and Databases*, Berlin, Germany (2003)
7. Brian, M.: Jena: Implementing the rdf model and syntax specification. Technical report, Hewlett Packard Laboratories, Bristol, UK (2000), <http://www.hpl.hp.com/semweb/index.html>
8. Cimiano, P.: *Ontology Learning and Population from Text-Algorithms, Evaluations and Applications*. Springer, Berlin, Heidelberg, Germany, Originally published as PhD Thesis, Universitt Karlsruhe (TH), Karlsruhe, Germany (2006)
9. David, V., Ivn, C., Miriam, F., Pablo, C.: A Multi-Purpose Ontology-Based Approach for Personalized Content Filtering and Retrieval. In: *Ist International Workshop on Semantic Media Adaptation and Personalization*, Athens, Greece, pp. 19–24 (2006)
10. Gruber, T.: <http://www-ksl.stanford.edu/kst/what-is-an-ontology.html>
11. Kemafor, A., Amit, S.: ρ -Queries: Enabling Querying for Semantic Associations on the Semantic Web. In: *WWW 2003*, Budapest, Hungary (2003)
12. Kemafor, A., Angela, M., Amit, S.: SPARQ2R: Towards Support for Subgraph Extraction Queries in RDF Databases. In: *WWW 2007*, Banff, Alberta, Canada (2007)
13. Krys, J., Macie, J.: SPARQLeR: Extended Sparql for Semantic Association Discovery. In: *Franconi, E., Kifer, M., May, W. (eds.) ESWC 2007. LNCS, vol. 4519*, pp. 145–159. Springer, Heidelberg (2007)
14. Li, D., Tim, F., Anupam, J.: Analyzing Social Networks on the Semantic Web. *IEEE Intelligent Systems* 8(6) (2004)
15. Prudhommeaux, E., Seaborne, A.: SPARQL Query Language for RDF. W3C Working Draft (2006), <http://www.w3.org/TR/2006/WD-rdf-sparql-query-20061004/>
16. Xuan, T., HaiHua, L., Xiaoyong, D.: Measuring Semantic Association in Domain Ontology. In: *International Conference on Semantics, Knowledge and Grid*, Xian, China (2007)
17. Xuan, T., Xiaoyong, D., HaiHua, L.: Computing Degree of Association Based on Different Semantic Relationships. In: *18th International Workshop on Database and Expert Systems Applications*, Regensburg, Germany (2007)
18. Yolanda, B., Jose, J., Pazos, A., Martin, L., Alberto, G.: AVATAR: An Improved Solution for Personalized TV based on Semantic Inference. *IEEE Transactions on Consumer Electronics* 52(1), 223–231 (2006)
19. Yong, H., Se, P., Seong, P., Young, L., Kweon, K.: Time Variant Event Ontology for Temporal People Information. *Fuzzy Logic and Intelligent Systems* 7(4), 301–306 (2007)
20. Zhdanova, A., Predoiu, L., Pellegrini, T., Fensel, D.: A Social Networking Model of a Web Community. In: *10th International Symposium on Social Communication*, Santiago, Cuba (2007)

Anomaly Detection over Spatiotemporal Object Using Adaptive Piecewise Model

Fazli Hanapiah, Ahmed A. Al-Obaidi, and Chee Seng Chan

Centre of Multimodal Signal Processing
Mimos Berhad,
Technology Park Malaysia,
57000 Kuala Lumpur, Malaysia
{fazli.hanapiah,ahmed.bahaa,cs.chan}@mimos.my
www.mimos.my

Abstract. Motion trajectories provide rich spatio-temporal information about an object activity. In this paper, we present a novel anomaly detection framework to detect anomalous motion trajectory using the fusion of adaptive piecewise analysis and fuzzy rule-based method. That is, first of all we address the problem by segmenting our moving objects using a Gaussian mixture background model. Secondly, visual tracking using probabilistic appearance manifolds to extract spatio-temporal trajectory. Thirdly, adaptive piecewise analysis and data quantization are performed on the extracted trajectory such that the anomalous detection can be performed as the incoming data are acquired. Finally, through the accumulative rank of the adaptive piecewise analysis and a fuzzy rule-based anomaly detection framework to detect the anomalous trajectory. Experimental results on various challenging trajectory data has validated the effectiveness of the proposed method.

1 Introduction

Detecting anomalous patterns from video sequence is useful for many applications such as surveillance, novelty extraction, automatic inspection and etc. The identification of anomalies can lead to the discovery of truly novel information from the video [12,5,2,13]. For instance, anomaly behaviour might be a person walking in a region not used by most people, a car following a zigzag path, or a person running in a region where most people simple walk. A *path* is any established line of travel or access, and a *trajectory* can be defined as a path followed by an object moving through the space.

In this paper, we present a framework for detecting nonconforming trajectories of objects as they pass through a scene by the fusion of adaptive piecewise analysis and fuzzy rule-based method. We concern ourselves primarily with human movements in a car park scene but the method is general and can be extended to any similar scenario. First of all, the moving objects in the image sequences are segmented using a Gaussian mixture background model; follow by the visual tracking using probabilistic appearance manifolds [9] to extract spatio-temporal

trajectory. Secondly, adaptive piecewise analysis and data quantization are performed on the extracted trajectory. That is, the adaptive piecewise analysis is performed after a sufficient amount of tracking data has been accumulated. The appropriate duration τ depends on the amount of the traffic in the scene and the required accuracy of the model. Finally, through the accumulative rank of adaptive piecewise analysis and a fuzzy rule-based anomaly detection framework to detect the anomalous trajectory.

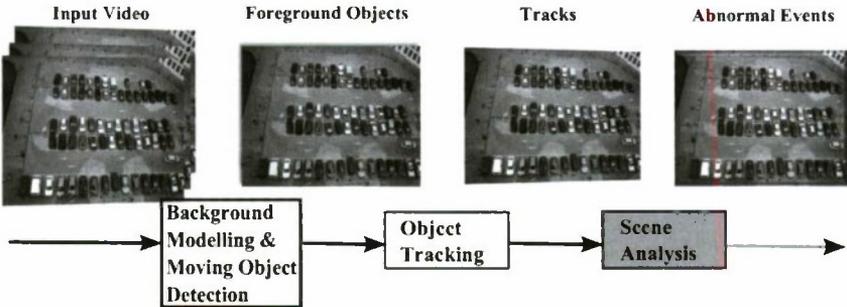


Fig. 1. The Proposed Anomalous Motion Trajectory Detection Framework

The advantages for the proposed are two-fold: on one hand, we would like to keep the problem computationally tractable where exhaustive training data and learning process can be avoided; on another hand, this provides a means to detect suspicious tracks through the accumulative rank of adaptive piecewise analysis and fuzzy rule-based method such that the anomalous detection can be performed as the incoming data are acquired, in opposition to off-line approaches like many of the aforementioned works. Our main aim is to avoid the classical two-step approaches (data collection and off-line processing).

The rest of the paper is structured as follows. Section 2 discuss the related work. Section 3 presents the proposed anomalous trajectory detection using the fusion of fuzzy rule and adaptive piecewise analysis. Section 4 shows the experimental results. Section 5 concludes the paper with discussions and future work.

2 Related Work

Trajectory analysis is an important step in applications like video surveillance, automotive systems, medical screening and autonomous robotic systems. Previous research on abnormal activity detection can be roughly divided into two categories: parametric approaches and non-parametric approaches. Grimson et. al. [6] use a distributed system of cameras to cover a scene, and employ an adaptive tracker to detect moving objects. Tracks are clustered using spatial features

on the vector quantisation approach. Once these clusters are obtained the unusual activities are detected by matching incoming trajectories to these clusters. Hu et al. [8] present a recently published technique in which the tracks are spatially and temporally clustered into different motion patterns. Each of these motion patterns is divided into several segments; each segment is modelled by a Gaussian model of speed and size. Makris and Ellis [4] develop a spatial model to represent the routes in an image. A trajectory is matched with routes already existing in a database using a simple distance measure. If a match is found, the existing route is updated by a weight update function; otherwise a new route is created for the new trajectory. One limitation of this approach is that only spatial information is used for trajectory clustering and behaviour recognition.

Another popular technique for activity recognition is Bayesian networks [3,1,11,7,14]. In [7], supervised training using Bayesian formulation is used for estimating the parameters of a multi-layered finite state machine model that is proposed for activity recognition. Very recently, Bayesian framework has been used for action recognition using ballistic dynamics [15]. This method is based on psycho kinesiological observations, that is, on the ballistic nature of human movements. Despite the fact that all these approaches have demonstrated success in modelling and recognizing the activities, all these methods need to have a large number of training sequences with intensive training in order for each activity to be recognised correctly which is not feasible for a real-time application.

3 Our Approach

Given a collection of unlabeled videos, we focus on the problem of interpreting the output of the object detection and tracking module in order to detect suspicious motion patterns. The proposed approach is illustrated in Figure 1.

3.1 Object Detection and Tracking

The visual tracking information serves as the input for our framework and we have employed the object detection and tracking system presented in [9]. The whole system includes the following component: a Gaussian mixture background model, motion detection from background subtraction and the appearance manifold based tracking algorithm to extract the trace of each object. The output of the tracker produces a set of m tracks $\{T_1, \dots, T_i, \dots, T_m\}$, where every track is a set of observation of the same object. For instance, any i^{th} track is a set of observations $T_i = \{O_1, \dots, O_j, \dots, O_n\}$, where $O_j = (x_t, y_t)$ contains the displacement of an object in the image plane (x, y) .

3.2 Adaptive Piecewise and Data Quantization

Piecewise linear analysis is ubiquitous. Let us model any nonlinear unicomparative function $g(f)$ with a constrained piecewise linear function $g_{PL}(f)$. A

piecewise function is defined with N linear segments over the interval $[x_0, x_N]$ as

$$g_{PL}(f) = \begin{cases} \gamma_1(f) & x_0 \leq f \leq x_1 \\ \gamma_2(f) & x_1 \leq f \leq x_2 \\ \vdots & \\ \gamma_N(f) & x_{N-1} \leq f \leq x_N \end{cases} \quad (1)$$

where $\gamma_n(f) = a_n f + b_n, n = 1, \dots, N$ is a linear segment and x_j represents $N + 1$ prespecified knots in $[x_0, X_N]$.

Given M pairs of samples $(f_m, g_m), m = 1, \dots, M - 1$ from the image sequences, the best-fit piecewise linear function to be the one that minimizes the cost function

$$J = \sum_{m=0}^{M-1} (g_m - g_{PL}(f_m))^2 \quad (2)$$

To minimize this cost function with respect to the $N - 1$ unknown parameters a_1, \dots, a_{N-1} , we can evaluate $\frac{\partial J}{\partial a_j} = 0$ where $j = 1, \dots, N - 1$ to get a system of $N - 1$ simultaneous equations in $N - 1$ unknowns:

$$\sum_{m=0}^{M-1} g_{PL}(f_m) \frac{\partial g_{PL}(f_m)}{\partial a_j} = \sum_{m=0}^{M-1} g_m(f_m) \frac{\partial g_{PL}(f_m)}{\partial a_j} \quad (3)$$

where note that

$$\frac{\partial g_{PL}(f_m)}{\partial a_j} \begin{cases} 0 & n < j \\ f_m - x_{j-1} & n = j \\ x_j - x_{j-1} & j < n < N \\ (1 - h(f_m))(x_j - x_{j-1}) & n = N \end{cases} \quad (4)$$

and $n = 1, \dots, N$.

However when identifying constant intervals a posteriori from a piecewise linear model, we risk mis-identifying constant intervals a posteriori from a piecewise linear model. In this paper, we adopted the adaptive piecewise analysis [10] (Algorithm 1) where we first apply the aforementioned piecewise linear analysis and then we seek to split the linear intervals into constant intervals (please refer to Fig. 2). That is, the algorithms only splits an interval if the fit error can be reduced, it is guaranteed not to degrade the fit error.

In this paper, we define a segmentation as a sorted set of segmentation indexes z_0, \dots, z_k such that $z_0 = 0$ and $z_k = n$. The segmentation points divide the time series into intervals S_1, \dots, S_k defined by the segmentation indexes as $S_j = \{(x_t, y_t) | z_{j-1} \leq t \leq z_j\}$. The segmentation error is computed from

$$\sum_{j=1}^k Q(S_j) \quad (5)$$

where function Q is the square of the Euclidean, l_2 regression error. Formally,

$$S_j = \min_p \sum_{r=z_{j-1}}^{z_j-1} (p(x_r) - y_r)^2 \quad (6)$$

Input: Time series (x_i, y_i) of length n
Input: Bound on polynomial degree N and model complexity k
Input: Function $E(p, q, d)$ computing fit error with poly in range $[x_p, x_q)$
 S empty list
 $d \leftarrow N - 1$
 $S \leftarrow (0, n, d, E(0, n, d))$
 $b \leftarrow k - d$
while $b - d \geq 0$ **do**
 find tuple (i, j, d, ϵ) in S with maximum last entry
 find minimum of $E(i, l, d) + E(l, j, d)$ for $l = i + 1, \dots, j$
 remove tuple (i, j, ϵ) from S
 insert tuples $(i, l, d, E(i, l, d))$ and $(l, j, d, E(l, j, d))$ in S
 $b \leftarrow b - d$
end
for tuple (i, j, q, ϵ) in S **do**
 find minimum m of $E(i, l, d') + E(l, j, q - d' - 1)$ for $l = i + 1, \dots, j$ and
 $0 \leq d' \leq -1$
 if $m < \epsilon$ **then**
 remove tuple (i, j, q, ϵ) from S
 insert tuples $(i, l, d', E(l, j, d'))$ and $(l, j, q - d' - 1, E(l, j, q' - 1))$ in S
 end
end

Algorithm 1. Adaptive Piecewise Algorithm

where the minimum is over the polynomials p of a given degree. For instance, if the interval S_j is said to be constant, therefore

$$Q(S_j) = \sum_{z_j \leq l \leq z_{j+1}} (y_l - \bar{y})^2 \quad (7)$$

where \bar{y} is the average, $\bar{y} = \sum_{z_{j-1} \leq l \leq z_j} \frac{y_l}{z_{j+1} - z_j}$. Similarly, if the interval has a linear model, then $p(x)$ is chosen to be linear polynomial $p(x) = ax + b$ where a and b are found by regression. The segmentation error can be generalised to other norms, such as the maximum-error (l_∞) norm [10,32] by replacing the \sum operator by max operators.

3.3 Data Quantization

In this paper, the adaptive piecewise analysis is performed after a sufficient amount of tracking data has been accumulated. The appropriate duration, τ depends on the amount of the traffic in the scene and the required accuracy of the model. For instance, the adaptive piecewise analysis m_τ is obtained from the observation vector $O_{j \rightarrow j+\tau}$ and empirically, we have chosen $\tau = 7$. Following this, a data-quantization process to represent the outcome qualitatively is conducted as to Eq. 8. An example of the process is illustrated in Fig. 3.

$$\begin{cases} m \leq 0 & 0 \\ \text{else} & 1 \end{cases} \quad (8)$$

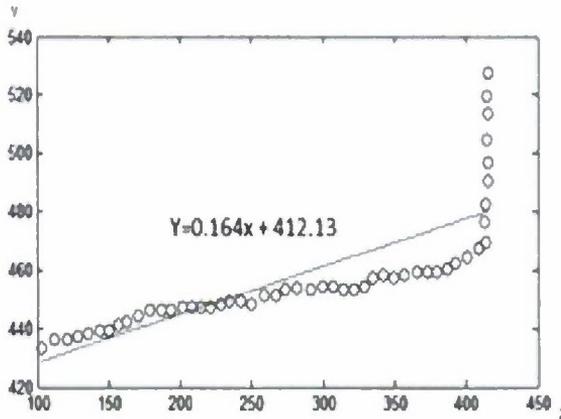
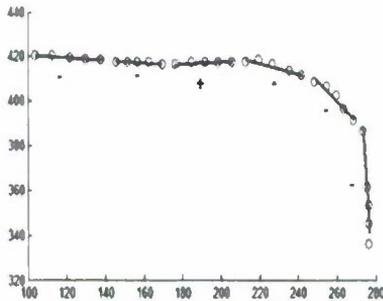
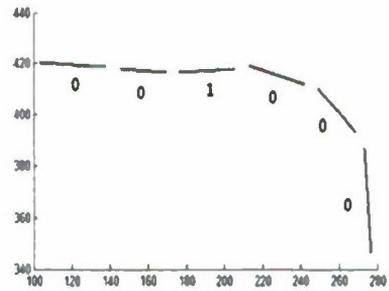


Fig. 2. Example of the Adaptive Piecewise Analysis



(a) Local Adaptive Piecewise Analysis using



(b) Data Quantization

Fig. 3. Adaptive Piecewise Analysis and Data Quantization

3.4 Fuzzy Rule-Based Anomaly Detection

In order to detect the anomalous motion trajectory, we propose a fuzzy rule-based system as to Fig. 4, which can automatically detect suspicious trajectories moving in atypical paths. Each feature (e.g. time and continuity) is passed through a set of fuzzy membership functions to get membership values corresponding to LOW, MEDIUM or HIGH, and finally the proposed fuzzy rules. In our proposed approach, we do not need the whole trajectory to perform the anomaly detection. As mentioned in previous section in this paper, anomaly detection is performed after a sufficient amount of tracking data has been accumulated.

The fuzzy inference engine consists of 6 rules where each of the rules will generate a response, corresponding to 'Very Usual', 'Usual', 'Usual or Suspicious', 'Suspicious' and 'Very Suspicious'. The output membership function corresponding to each of these responses is shown in Fig. 5.

1. If Time is **HIGH** and Continuity is **LOW** then *USUAL OR SUSPICIOUS*
2. If Time is **HIGH** and Continuity is **MEDIUM** then *USUAL*
3. If Time is **HIGH** and Continuity is **HIGH** and then *VERY USUAL*
4. If Time is **LOW** and Continuity is **LOW** then *USUAL OR SUSPICIOUS*
5. If Time is **LOW** and Continuity is **MEDIUM** then *SUSPICIOUS*
6. If Time is **LOW** and Continuity is **HIGH** then *VERY SUSPICIOUS*

where

- Time = The difference between $Time_i$ and $Time_{i+\tau}$
- Continuity = The similarity of the piecewise linear analysis between $Time_i$ and $Time_{i+\tau}$

Fig. 4. The Fuzzy Rules

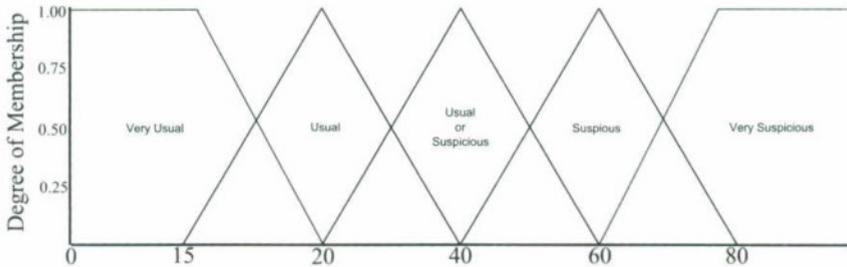


Fig. 5. Output Membership Functions

4 Experiments

In this section, we present the effectiveness of the proposed approach into detecting 100 datasets which consist of both benign (normal) and potentially dangerous (suspicious) categories. The validation scenario is an outdoor environment, such as a parking lot.

4.1 Experimental Setup

First, the trajectory of a moving object is extracted from the background image by subtracting the image of the tracked object with the background image models by Gaussian mixture. The trajectory obtained can be given as follow: $T_i = \{O_1, \dots, O_j, \dots, O_n\}$, where $O_j = (x_t, y_t)$. Using the x-y coordinate points, we perform the piecewise linear based on different duration. Next, we use the gradient information, m from the piecewise linear analysis to produce qualitative data. Our condition for the quantization process is that if $m \leq 0$ (positive) then it will be represented as '1' and else it will be represented as '0'. Finally, with the

```

Input: Trajectory  $\{(x_i, y_i)\}_{i=1, \dots, t}$ 
for Time,  $T = 1 \longleftarrow T_{end}$  do
  CHECK  $T_i == T_{i+1} ??$ 
  if YES then
    CHECK CONTINUOUS triggered ??
    while YES do
      Suspicious behaviour indication (SBI) =  $SBI_{current} - 20\%$ 
    else
      SBI = 50%
      SET CONTINUOUS = 1
    end
  end
end
if NO then
  CHECK NONCONTINUOUS triggered ??
  while YES do
    SBI =  $SBI_{current} + 20\%$ 
    SET CONTINUOUS = 0;
  else
    SBI = 50%
    SET CONTINUOUS = 0
    SET NONCONTINUOUS = 1
  end
end
end
return SBI for each  $\{(x_i, y_i)\}_{i=1, \dots, t}$ 

```

Algorithm 2. Anomaly Detection Algorithm

qualitative data, anomaly trajectory is detected by comparing to the proposed rule-based framework.

4.2 Results and Discussions

Experiments are conducted to test the effective of the approach in detecting the anomaly path trajectory using proposed fuzzy rule-based framework. The overall performance of the method is tested against 50 normal trajectories, and 50 suspicious trajectories. The results are shown in the Table 1.

Based on these results, the proposed approach manages to give the accuracy up to 90% depending on the frame rate are used. The choice of frame rate is empirically chosen. This result is considered as good compare to other approaches which required extensive training data set and offline learning process which is computationally expensive. Furthermore, we also closely examined the misclassified trajectories for each windows size and noticed that most of the misclassified trajectories were found in the same tracked objects, T_i (please refer to Fig. 6). One of the main reasons is due to the distorted trajectories points (noise) during the tracking process. We felt that this problem can be alleviated by using more accurate tracker and this is work in progress.

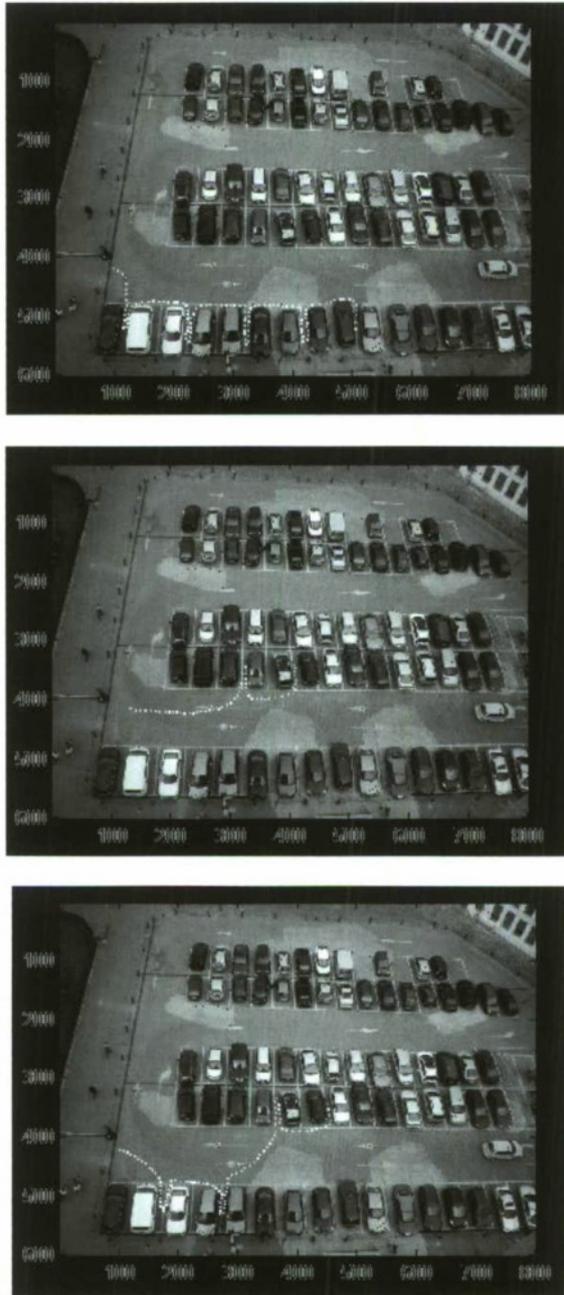


Fig. 6. Sample of the Abnormal Trajectories Dataset

Table 1. Classification Results with Different Frame Rate

# of Subjects		Accuracy for Frame Rate (<i>fps</i>)		
		6	12	25
Single Person	Normal Trajectory Correctly Classified	96%	93%	95%
	Suspicious Trajectory Correctly Classified	89%	84%	88%
Multiple Person	Normal Trajectory Correctly Classified	93%	94%	93%
	Suspicious Trajectory Correctly Classified	90%	83%	88%
Total Accuracy		92%	90%	91%

Correlation Analysis. In the second approach, we only consider the correlation analysis to perform anomaly detection as a comparison. Correlation is to measure the closeness of the linear relationship between X and Y after the regression process. In this paper, we employed the Pearson’s product-moment correlation coefficient to measure this linear relationship (Eq. 9).

$$R = \frac{n \sum x_n y_n - \sum x_n \sum y_n}{\sqrt{n \sum x_n^2 - (\sum x_n)^2} \sqrt{n \sum y_n^2 - (\sum y_n)^2}} \tag{9}$$

The value for correlation coefficient R can be varied from 1 to -1 depending on the data.

- $R = 0$ is no linear correlation
- $R = 1$ is perfect +ve linear correlation (+ve gradient)
- $R = -1$ is perfect -ve linear correlation (-ve gradient)

From this measure, we calculated the deviation of the path from the obtained regression line. This would mean that if the object tracked is in a normal path trajectory the R value will be relatively close to 1 or -1 and if the object tracked is in the abnormal path trajectory the R will be closer to 0. Thirty normal path trajectories have been analysed and the correlation coefficient are shown in Table 2. From the Table 2, it can be noticed that the R range is lied between $0.91 < R < 0.53$ for positive correlation and $-0.52 < R < -0.82$ for negative correlation. Same approach is used to analyse nine abnormal path trajectories and the results are Table 3. R range for abnormal path trajectories are $0.85 < R < 0.22$ and $-0.12 < R < -0.87$. However, these results do not give any significant correlation value to distinguish between normal and abnormal path trajectories as the correlation range for abnormal path are overlap with the normal path correlation range.

Table 2. Correlation Coefficient, R for Each Normal Trajectory Dataset

Dataset, S_i	Correlation Coefficient, R	Dataset, S_i	Correlation Coefficient, R	Dataset, S_i	Correlation Coefficient, R
1	0.7703	11	0.5913	21	-0.4666
2	0.8464	12	0.7884	22	-0.6628
3	0.8592	13	0.6049	23	-0.7293
4	0.9105	14	-0.8239	24	-0.6131
5	0.8426	15	-0.7176	25	-0.6285
6	0.8128	16	-0.5652	26	-0.5104
7	0.8181	17	-0.6895	27	0.9119
8	0.7953	18	-0.6767	28	0.7566
9	0.5371	19	-0.5910	29	-0.6651
10	0.6832	20	-0.6949	30	-0.7339

Table 3. Correlation Coefficient, R for Each Abnormal Trajectory Dataset

Dataset, S_i	Correlation Coefficient, R	Dataset, S_i	Correlation Coefficient, R	Dataset, S_i	Correlation Coefficient, R
1	0.2212	4	0.8550	7	-0.6886
2	-0.7137	5	-0.8695	8	0.4593
3	-0.6278	6	-0.7896	9	-0.1164

5 Concluding Remarks

In this paper, we presented the hybrid adaptive piecewise linear-fuzzy rule-based anomalous trajectory detection algorithms and experimental results using various challenging trajectories has validated the proposed method. Our aim in this presentation has been to motivate the need for, and challenges involved in, the detection of anomalous temporal data resulting from object tracking captured. The proposed algorithm is significant over the state-of-the art methods in a way that 1)no extensive training and learning are required and 2)the anomaly detection is performed as the incoming data are acquired, therefore avoid the classical two-step approaches (data collection and off-line processing). Our future work will focus on automatically extracting the rules explaining the phenomena hidden into the input data, for trajectory analysis and introduce the interactions between objects to the trajectory patterns.

References

1. Brand, M., Oliver, N., Pentland, A.: Coupled hidden markov models for complex action recognition. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 994–1002 (1997)
2. Chan, C.S., Liu, H.: Fuzzy qualitative human motion analysis. IEEE Transactions on Fuzzy Systems 17(4), 851–862 (2009)

3. Cuntoor, N., Yegnanarayana, B., Chellappa, R.: Activity modeling using event probability sequences. *IEEE Transactions on Image Processing* 17(4), 594–607 (2008)
4. Dimitrios, M., Ellis, T.: Path detection in video surveillance. *Image and Vision Computing* 20(12) (2002)
5. Duong, T., Bui, H., Phung, D., Venkatesh, S.: Activity recognition and abnormality detection with the switching hidden semi-markov model. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 838–845 (June 2005)
6. Grimson, W., Stauffer, C., Romano, R., Lee, L.: Using adaptive tracking to classify and monitor activities in a site. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 22–28. IEEE Computer Society, Los Alamitos (1998)
7. Hongeng, S., Nevatia, R.: Multi-agent event recognition. In: *Proceedings of the Eighth IEEE International Conference on Computer Vision*, vol. 2, pp. 84–91 (2001)
8. Hu, W., Xiao, X., Fu, Z., Xie, D., Tan, T., Maybank, S.: A system for learning statistical motion patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28(9), 1450–1464 (2006)
9. Lee, K.-C., Ho, J., Yang, M.-H., Kriegman, D.: Visual tracking and recognition using probabilistic appearance manifolds. *Computer Vision and Image Understanding* 99(3), 303–331 (2005)
10. Lemire, D.: A better alternative to piecewise linear time series segmentation. In: *Proceedings of the SIAM International Conference on Data Mining* (April 2007)
11. Medioni, G., Cohen, I., Brmond, F., Hongeng, S., Nevatia, R.: Event detection and analysis from video streams. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23(8), 873–889 (2001)
12. Niu, W., Long, J., Han, D., Wang, Y.-F.: Human activity detection and recognition for video surveillance. In: *IEEE International Conference on Multimedia and Expo.*, vol. 1, pp. 719–722 (June 2004)
13. Oliver, N., Garg, A., Horvitz, E.: Layered representations for learning and inferring office activity from multiple sensory channels. *Computer Vision and Image Understanding* 96(2), 163–180 (2004)
14. Park, S., Aggarwal, J.: A hierarchical bayesian network for event recognition of human actions and interactions. *Multimedia Systems* 10(2), 164–179 (2004)
15. Vitaladevuni, S., Kellokumpu, V., Davis, L.: Action recognition using ballistic dynamics. In: *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8 (June 2008)

Experimental Analysis of the Effect of Dimensionality Reduction on Instance-Based Policy Optimization

Hisashi Handa

Okayama University, Okayama 700-8530, Japan

handa@sdc.it.okayama-u.ac.jp

<http://www.sdc.it.okayama-u.ac.jp/~handa/>

Abstract. Manifold Learning has attracted much attention for this decade. One of the main features of Manifold Learning is that Manifold Learning tries to conserve local topologies in high-dimensional space. In this paper, we discuss the effect of the dimensionality reduction of input spaces of Evolutionary Learning. We examine two Manifold Learning algorithms: Isomap and LLE. We adopt the Instance-Based Policy Optimization as an Evolutionary Learner. In addition, we introduce a metric of relative error of distances between original input space and reduced space. We will show the relationship between this metric and the number of neighbors in Manifold Learning.

1 Introduction

In this study, we investigate the effect of the dimensionality reduction in Evolutionary Learning. In evolutionary learning, the alignment of sensors is a key issue to design effective intelligent agents/robots. It is impossible to solve problems with insufficient sensor information while redundant sensory inputs causes considerable amount of learning time.

In this paper, Isomap or LLE (Locally Linear Embedding), one of Manifold Learning Algorithms, is used to reduce the number of dimensionality of sensory inputs [1,2]. By using the reduced inputs, agents decide their actions and learn policies to achieve a given task. An important feature of the Manifold Learning Algorithms is to preserve local topological relationship among data. Fig. 1, for instance, depicts the S-shaped data, which is often used to explain the effectiveness of the Manifold Learning. The left graph in this figure denotes original data in a three-dimensional space, which are sampled from a two-dimensional manifold. Note that colors of points have no special meanings. They are just for ease of understandings. The right graph in the figure is a typical result by Manifold Learning for the original data. The order of color sequence is maintained in this resultant two dimensional data.

In this paper, we propose a two-stage learning method for mobile robots: The first stage is to learn the mapping from high dimensional sensory inputs to low dimensional data. The high dimensional sensory inputs are collected by the

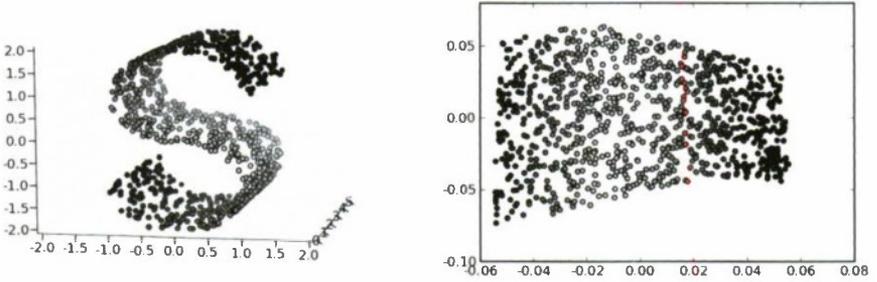


Fig. 1. S-shaped data (LEFT) and the typical result by Manifold Learning (RIGHT)

elopement in another environment of tasks where there are various kinds of obstacles. The Manifold learning is used to generate the low dimensional data. The second state is to learn the policy of robots by using Evolutionary Algorithms. At every time step, the robots perceive the high dimensional sensory inputs as the same as in the first stage. Then, the low dimensional data associated with perceived the high dimensional data is given to an individual in the Evolutionary Algorithms. This paper examines various combinations of parameters for IBP (Instance-Based Policy Learning) with dimension reduction algorithms. Especially, we investigate the relationship between relative errors in dimension reduction, and the number of neighbors k . In addition, we compare the proposed method with evolutionary learning with hand-tuned sensors.

Related works are described as follows: Dimension reduction techniques including SOM are often used in conventional reinforcement learning community and as genetic operations or visualization tools of individuals in Evolutionary Optimization [3,4,5,6,7]. In the case of Evolutionary Learning, there is few research. We can guess some reasons of this: One of main stream of applying Evolutionary Learning to robotics is of Learning Classifier Systems (LCS) [8]. In the case of LCS, schemata are quite important notion of them. If we use dimension reduction techniques, it would be difficult to constitute effective schemata. Another evolutionary approach is use of Neural Networks, i.e., NeuroEvolution [9]. In this case, they would rely on the information processing capability of Neural Networks for non-linear phenomena. In robotics, Manifold Learning have attracted much attention for generating Maps [10]. Our research can be regarded as an extension of this study to Evolutionary Learning.

2 Manifold Learning

The first generation of Manifold Learning algorithms, i.e., Locally Linear Embedding and Isomap, is proposed in 2000 [1,2,11]. These have attracted much attention especially in image processing community since these can embed the relationship among a large number of images into two dimensional space naturally. Hence, several subsequent algorithms have been proposed such as Laplacian

Eigenmaps, Hessian Eigenmaps, and so on. In this paper, in order to investigate the effectiveness of the information processing on Manifolds, we employ basic Manifold Learning Methods, i.e., Isomap, and LLE.

2.1 Locally Linear Embedding

The LLE algorithm tries to maintain the local topology in reduced space. As mentioned below, the LLE algorithm is based on linear algebra for calculating the positions in the reduced space while it can achieve highly nonlinear embeddings. The LLE algorithms is executed as follows [11]:

1. Assign neighbors to each data point \mathbf{x}_i .
2. Compute the weights w_{ij} that best linearly reconstruct \mathbf{x}_i from its neighbors, by solving this equation:

$$\epsilon(w) = \sum_i |\mathbf{x}_i - \sum_j w_{ij} \mathbf{x}_j|^2.$$

3. Compute the low-dimensional embedding vectors \mathbf{y}_i by using the weights w_{ij} and the following equation:

$$\Phi(Y) = \sum_i |\mathbf{y}_i - \sum_j w_{ij} \mathbf{y}_j|^2.$$

2.2 Isomap

Isomap, proposed by Tenenbaum *et al.* is one of the most famous Manifold Learning Algorithms [1]. In the Isomap, the geodesic distance on Manifolds is used instead of the Euclidean distance. The procedure of the Isomap is described as follows:

1. K-Nearest Neighbor method is adopted all the input data x_i . Then, a neighborhood graph x is constructed such that nodes in the graph is connected if they are of neighbor in the sense of K-Nearest Neighbor method. Distance $d_G(i, j)$ of edge among connected nodes is set to be $d_x(i, j)$, i.e., Euclidean distance between the input data x_i and x_j .
2. For all the pair s x_i, x_j of input data, the shortest path distance $d_G(i, j)$ on the neighborhood graph G are calculated.
3. A low dimensional projection is generated by calling a metric MDS (Multi Dimensional Scaling) and by using the the shortest path distance $d_G(i, j)$.

3 Instance-Based Policy Learning

The instance based policy learning proposed by Miyamae is an evolutionary approach for solving reinforcement learning problems [12]. It is composed of several vectors, called instances. Each instance consists of a state part and an action part. For a given perceptual input at each time step, the nearest instance

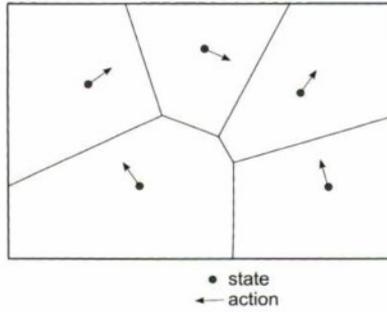


Fig. 2. Example of an Individual: Instances in perceptual input space

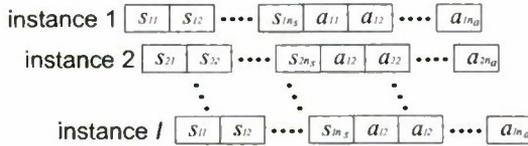


Fig. 3. Representation of Individuals in Instance Based Policy learning

is activated as nearest neighbor method. The action of the activated instance is taken. Fig. 2 depicts an example of instances in perceptual input space, where the dimensions of states and actions are 2 and 1, respectively. The position of circles and the orientation of arrows denote the state part and the action part of instances, respectively. As delineated in the figure, the perceptual input space is segmented into several subspaces as in Voronoi diagrams. Each subspace is associated with one of instance. That is, each of instance activates for perceptual inputs in a corresponding subspace. The arrows in the figure illustrate actions for corresponding instances.

Fig. 3 describes the genotype for the Instance Based Policy learning. s_{ij} and a_{ik} denote the j^{th} element of state vector and the k^{th} element of action vector of i^{th} instance. I indicates the number of instances, which is predefined. n_s and n_a represents the number of states and actions, respectively. All the variables s_{ij} and a_{ik} are represented by a real value. Hence, any Evolutionary Algorithms for continuous function optimization problems can be used. This paper utilizes CMA-ES (Covariance Matrix Adaptation Evolution Strategies) while the original paper of the IBP learning method uses the Real-Coded GA proposed by their research group for evolution [12,13]. The reason of the utilization is due to the availability of the source code. We believe there is no significant difference between the CMA-ES and the Real-Coded GA since we do not have to find out the optimal policies with high degree of precision as in ordinal function optimization problems.

4 Proposed Method

A two-stage learning method for mobile robots is proposed in this paper as depicted in Fig. 4. The first stage is to constitute a mapping from sensory inputs \mathbf{x} to low dimensional data \mathbf{y} . The second stage is Evolutionary Learning to achieve a given task. In this stage, at every time step t , sensory inputs \mathbf{x}_t is transformed to corresponding input \mathbf{y}_t by using the mapping. Hence, the inputs for the learner is \mathbf{y}_t .

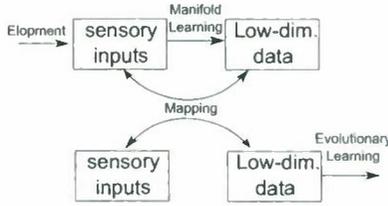


Fig. 4. Diagram of the proposed method

4.1 Constitution of Mapping

At first, data collection is carried: A robot moves in a given environment around. In this paper, we set up another environment for this elopement, where there are variety of obstacles. After a large number of sensory inputs are gathered, data with no activated sensors are eliminated. Moreover, a predefined number of data is randomly chosen from the eliminated data set.

The dimension reduction method is carried out for the chosen data. This paper examines not only Isomap but also Kernel PCA algorithms for this purpose [14]. The chosen data \mathbf{x}_i and the reduced data \mathbf{y}_i are associated, where $i = 1, \dots, n_d$, and n_d indicated the number of the chosen data.

4.2 Transformation of Sensory Inputs in Evolutionary Learning Phase

As mention above, at every time step t , sensory inputs \mathbf{x}_t should be transformed: Firstly, the nearest and the second nearest point $\mathbf{x}', \mathbf{x}''$ from the chosen data for \mathbf{x}_t is found out. Secondly, the current sensory inputs \mathbf{x}_t is projected to the line defined by two points $\mathbf{x}', \mathbf{x}''$. The projected point \mathbf{x}^* is regarded as the relative position α on the line as delineated in Fig. 5:

$$\alpha = \frac{\mathbf{x}^* - \mathbf{x}'}{\mathbf{x}'' - \mathbf{x}'}$$

The inputs \mathbf{y}_t for individuals are defined as follows:

$$\mathbf{y}_t = \alpha(\mathbf{y}'' - \mathbf{y}') + \mathbf{y}'$$

where \mathbf{y}' and \mathbf{y}'' are points in reduced space, which are associated with \mathbf{x}' and \mathbf{x}'' , respectively.

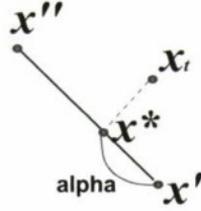


Fig. 5. Transformation of sensory inputs

In the case that the dimension of sensory inputs is high and the number of chosen data is large, it takes much time for finding out the nearest and the second nearest data \mathbf{x}' , \mathbf{x}'' . Locality-Sensitive Hashing (LSH) is used for finding such nearest points effectively [15].

5 Experiments

5.1 Configuration of Robots

We employ Simbad, a Java 3d robot simulator, for constructing simulated environment [16]. The mobile robot used in this paper is described as follows: The radius of the robot is 0.3 meters. 20 time steps per second are simulated. The robot has a large number of sonar sensors. We examined 72 or 12 sonar sensors for the proposed method. The allocation of these sensors are the same: The first sensor is set to be in the front of the robot. Other remaining sensor is equiangularly allocated, i.e., at every 5 degree for 72 sonar sensors. The range of sensors is 1.5 meters. The robot goes forward with 0.5 meters per second if there is no activated sensor. Otherwise the transformation and the rotational velocity of the robot is set to be 0.2 meters per second and $(a - 0.5) \times \pi/2$ meters per second, respectively, where a denotes the action of agent. The action a in this paper continuously varies from 0 to 1.

5.2 Dimension Reduction

Fig. 6 delineated the simulated environment for collecting variety of sensory inputs. The size of the field is 18 meters \times 18 meters. Various size of walls and blocks are stored. A robot with 72 sonar sensors moves in this field around for sufficient time. As mentioned in the previous section, data for dimension reduction is randomly chosen. The number of chosen data n_d is set to be 1000. We generate 10 kinds of datasets by using different random seeds for this choice. From this 1000 data for 72 sonar sensors, we generate other kinds of dataset by neglecting certain sensor values, i.e., data data for 12 sonar sensors.

Several mappings are generated by using Isomap and LLE. As mentioned above, we now have 2 kinds of datasets, where each dataset is composed of 10 sub-datasets with different random seeds. For each dataset, dimension reduction

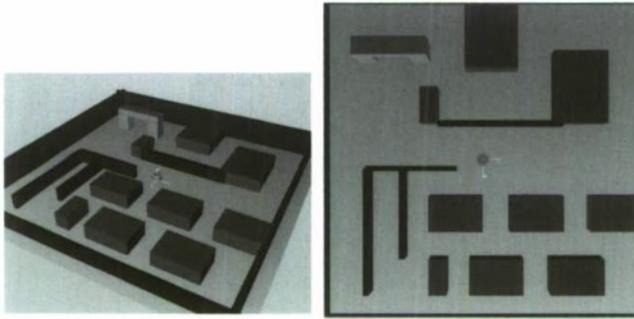


Fig. 6. Simulated environment for collecting a variety of sensory inputs

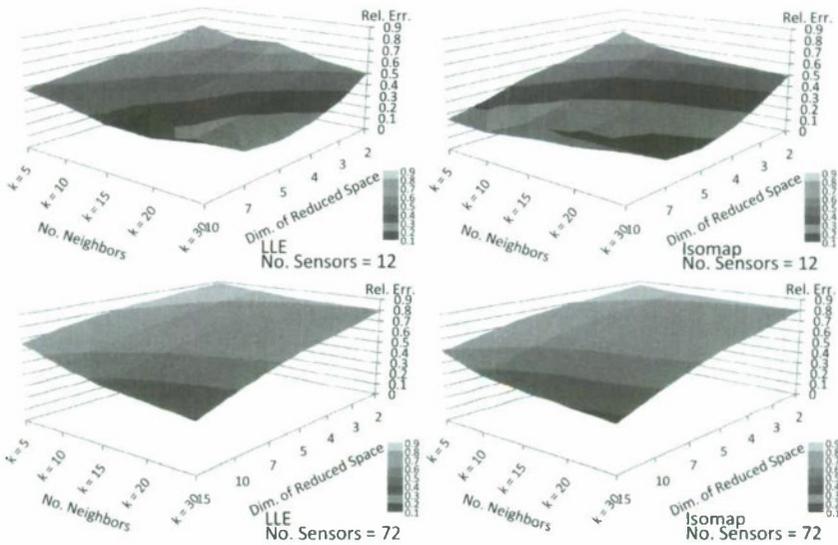


Fig. 7. Relative errors for various couples of the dimension of the reduced spaces and the number of neighbors: the upper and lower graphs are of that the number of sonar sensors are 12 and 72, respectively; LLE (LEFT), and Isomap (RIGHT)

method is carried out. The dimensions of reduced space are set to be 2, 3, 4, 5, 7, 10 and 15. In the case of the dataset for 12 sonar sensors, we did not apply the 15 dimensions of reduced space. In addition, we examined various numbers of neighbors k for Isomap and LLE. $k = 5, 10, 15, 20,$ and 30 are examined.

We introduce the relative error to evaluate the reduced space. This relative error is calculated as follows:

$$E(X, Y) = \sum_i^{n_d} \sum_{j=i+1}^{n_d} \frac{|D_g(\mathbf{x}_i, \mathbf{x}_j) - D_c(\mathbf{y}_i, \mathbf{y}_j)|}{D_g(\mathbf{x}_i, \mathbf{x}_j) n_d(n_d + 1)/2}$$

where $D_g(\cdot, \cdot)$ and $D_e(\cdot, \cdot)$ indicate the geodesic distance, estimated by Isomap, in original space, and the Euclid distance in the reduced space. Note that the LLE algorithm does not use the geodesic distance at all. However, the LLE uses the notion of neighbors so that this metric is also useful for the LLE. Fig. 7 depicts the relative errors for various couples of the dimension of the reduced spaces and the number of neighbors k . These values are averaged over 10 datasets which are randomly chosen with various random seeds. As increasing the number of neighbors, the corresponding relative errors are decreasing. In addition, as increasing the dimensions of the reduced spaces, the corresponding relative errors are also decreasing.

5.3 Experimental Results

We employ all the combinations of (algorithm, the number of sonar sensors, the dimension of reduced space, the number of neighbors k) as indicated in the previous subsection: The algorithm is either of LLE or Isomap. 12 or 72 sonar sensors are examined. These algorithms reduce the dimensions of original input space, which is equivalent to the number of sonar sensors, into 2, 3, 4, 5, 7, 10, or 15 dimensional space. For 12 sonar sensors, the reduction to 15 dimensional space is not carried out. k is set to be either of 5, 10, 15, 20 or 30. For comparison, Kernel PCA is also examined. The Kernel PCA does not use the notion of neighbors so that, except for k , similar combinations of parameters as mentioned above are examined.

Two simulated environments are examined as depicted in Fig. 8. In these depictions, a robot is located on the initial position. The goal area is located at the red line in the left side of these figures. 500 seconds (equivalent to 10,000 steps) are allowed to use for a single examination. The episode will be terminated if the robot reaches the goal, the robot bumps obstacles, or 500 seconds are exceeded.

The evaluation of a single examination is calculated as follows: The following function e is applied if a robot reaches to the goal.

$$e = 0.1 + 1.0/(\text{No. steps})$$

The second term in the right side of this equation is a very small number in comparison with the first term, i.e., 0.1. Therefore, the evaluation for success is almost 0.1 but it is greater if the robot could reach to goal faster. Otherwise, the evaluation is as follows:

$$e = -1.0/(\text{No. steps}) \times (\text{distance to the goal})$$

This evaluation is a very small negative number. The evaluation is worse if the robot bumps promptly or the robot could not get up close to the goal. For a single fitness evaluation, five examinations are carried out. The fitness function of individuals is calculated by the sum of five evaluations.

The number of instances in the Instance Based Policy optimization is set to be 5. n_s is the same as the dimensions of reduced space. n_a is 1, i.e., action a in the previous subsection. The length of individuals is $(n_s + n_a) *$ (the number of instances).

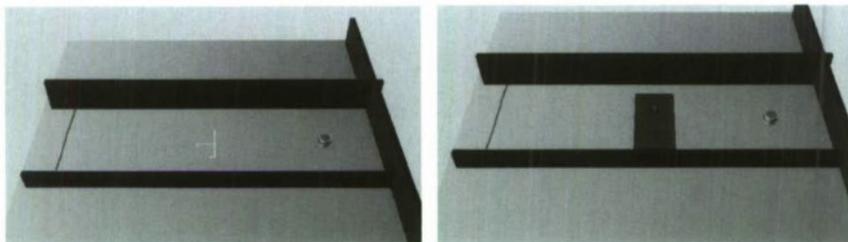


Fig. 8. Simulated Environments

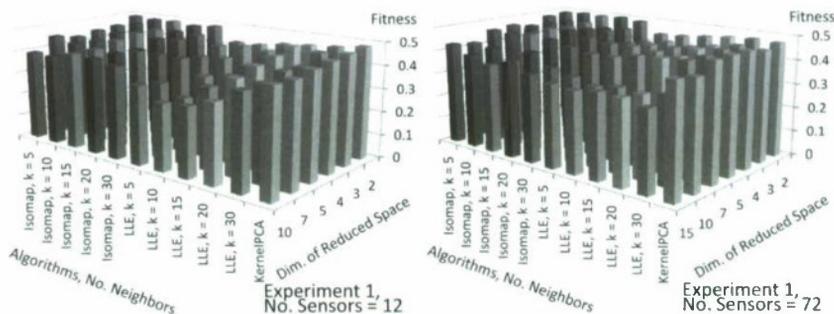


Fig. 9. Experimental results: Averaged Fitness after evolution; Simulated environments with no obstacles; (LEFT: the number of sensors is 12; RIGHT: 72)

Fig. 9 shows the experimental results for the simplest simulated environment, i.e., the left picture in Fig. 8. The left and right graphs mean that the number of sonar sensors are 12 and 72, respectively. The horizontal axis denotes the averaged fitness values after evolution for corresponding combinations of (algorithm, the number of neighbors k , the dimension of the reduced space). The best performances among them are (Isomap, 20, 2) for 12 sonar sensors, and (Isomap, 30, 2) for 72 sonar sensors. In the same parameters for algorithm and the dimension of the reduced space, as increasing k , the performance tends to increase. This tendency is similar to the relative error shown in subsection 5.2.

Fig. 10 shows the experimental results for the simulated environment with one obstacle, i.e., the right picture in Fig. 8. It is difficult to clearly see the tendency as in the result for the simulated environment without obstacles. However, in the case of Isomap, larger k causes better results. LLE does not work well for this environment. For 72 sonar sensors, performances are deteriorated for all the algorithms if the dimension of reduced spaces is around 5. As we can see in Fig. 7, the relative error is improved if the dimension of reduced spaces is increasing. However, such increase causes the performance deterioration at Evolutionary Learning phase. In the case of this simulated environment with 72 sonar sensors, there would be suboptimal at that the dimension of reduced spaces is 10.

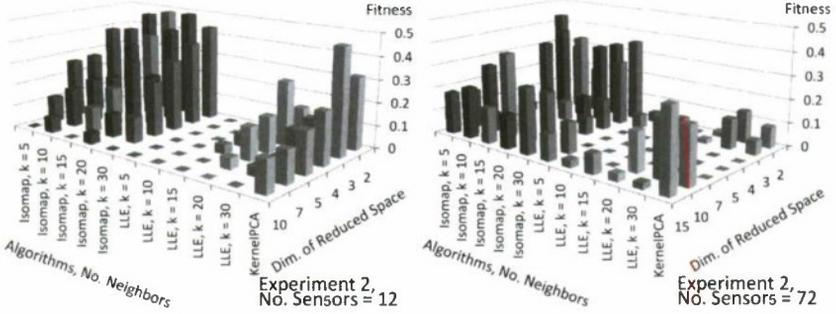


Fig. 10. Experimental results: Averaged Fitness after evolution; Simulated environments with one obstacle; (LEFT: the number of sensors is 12; RIGHT: 72)

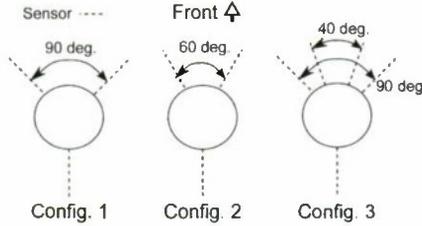


Fig. 11. Hand-Tuned sensor configurations for conventional IBP method

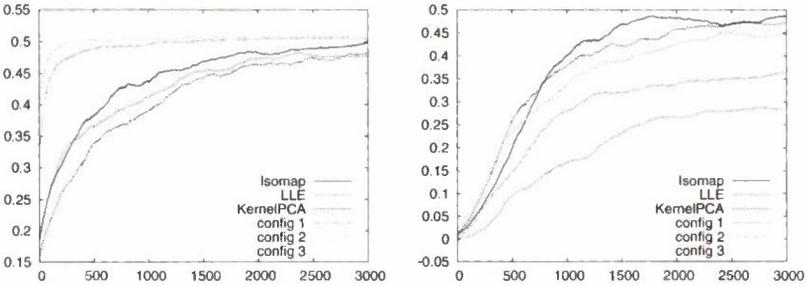


Fig. 12. Experimental results: The changes of fitness by IBP with hand-tuned sensors: Simulated environments without obstacles (LEFT); with one obstacle (RIGHT)

Finally, we compare the proposed methods with IBP with hand-tuned sensor allocations. We show here three sensor configurations as shown in Fig. 11. Fig. 12 shows experimental results of this comparison. The left and right graphs show the result of the simulated environment without obstacles and with one obstacle, respectively. In the simple environment, IBP with hand-tuned sensors can acquire optimal policy rapidly. On the other hand, the IBP with configuration 1 and 3 cannot solve the simulated environment with one obstacle well. The

IBP with configuration 2 works well while its performance is worse than the one of Isomap and Kernel PCA. These configurations may not be optimal one since we only examined over 20 configurations. These results, however, elucidate the difficulties of the allocation of sensors for general purpose.

6 Conclusions

In this paper, we examined various combinations of parameters for IBP with dimension reduction algorithms: two kinds of Manifold Learning algorithms, i.e., Isomap and LLE; the number of neighbors k ; the number of sensors, i.e., the dimension of original input space; the dimension of reduced spaces. We introduced the relative error to investigate how the dimension reduction worked well. For the number of neighbors, as the relative errors are decreasing, the fitness tends to be improved. However, in terms of the dimension of reduced spaces, such tendency could not be observed: The relative errors are decreasing if the dimension of reduced spaces is increasing. At the time, the performance is also deteriorated. One of this reason is that the length of individuals are growing in proportion to the dimension of reduced space.

In addition, we compared with IBP with hand-tuned sensors. This experiment reveal the difficulty of sensor allocations with several sensors for general purpose. That is, the proposed method can avoid such difficulty efficiently.

Future works are described as follows: The proposed method is two-staged algorithm, that is, batch-process is adopted. It would be better to apply on-line version of Manifold Learning for practical application. In this case, during evolution, the meanings of input value could be changed by Manifold Learning. Some isomorphism mechanisms should be devised. We may be able to incorporate the geodesic distance into Evolutionary Learning, instead of the use of Manifold Learning. In this case, we need to take account into the curse of dimensionality.

Acknowledgment

This work was partially supported by the Grant-in-Aid for Exploratory Research, the Grant-in-Aid for Scientific Research (B), and the Grant-in-Aid for Young Scientists (B) of MEXT, Japan (18656114, 21360191, and 21700254).

References

1. Tenenbaum, J.B., de Sliva, V., Lagford, J.C.: A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science* 290(22), 2319–2323 (2000)
2. Huo, X., Ni, X.S., Smith, A.K.: A survey of manifold-based learning methods. In: Liao, T.W., Triantaphyllou, E. (eds.) *Recent Advances in Data Mining of Enterprise Data*. World Scientific, Singapore (2007)
3. Tateyama, T., Kawata, S., Oguchi, T.: A teaching method using a self-organizing map for reinforcement learning. *Artificial Life and Robotics* 7(4), 193–197 (2006)

4. Handa, H., Ninomiya, A., Horiuchi, T., Konishi, T., Baba, M.: Adaptive State Construction for Reinforcement Learning and its Application to Robot Navigation Problems. In: Proc. 2001 IEEE Sys. Man and Cybernetics Conf., pp. 1436–1441 (2001)
5. Hiroyasu, T., Miki, M., Sano, M., Shimosaka, H., Tsutsui, S., Dongarra, J.: Distributed Probabilistic Model-Building Genetic Algorithm. In: Proc. 2003 Genetic and Evol. Comp. Conf., pp. 1015–1028 (2003)
6. Najafi, M., Beigy, H.: Using PCA to improve evolutionary cellular automata algorithms. In: Proc. 2009 Genetic and Evol. Comp. Conf., pp. 1129–1130 (2009)
7. Obayashi, S., Sasaki, D.: Visualization and Data Mining of Pareto Solutions Using Self-Organizing Map. In: Proc. 2nd Inter. Conf. on Evol. Multi-Criterion Opt., pp. 796–809 (2003)
8. Goldberg, D.E.: Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley, Reading (1989)
9. Miikkulainen, R., Stanley, K.: Evolving Neural Networks Through Augmenting Topologies. *Evolutionary Computation* 10(2), 99–127 (2002)
10. Nishizuka, K., Yairi, T., Machida, K.: Simultaneous Localization and Mapping of Mobile Robot Using Nonlinear Manifold Learning. In: Proc. 36th Inter. Sympo. on Robotics (2005)
11. Roweis, S.T., Saul, L.K.: Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science* 290(22), 2323–2326 (2000)
12. Miyamae, A., Sakuma, J., Ono, I., Kobayashi, S.: Instance-based Policy Learning by Real-coded Genetic Algorithms and Its Application to Control of Nonholonomic Systems. *Tran. Japanese Soc. for Artificial Intelligence* 24(1), 104–115 (2009)
13. Hansen, N., Ostermeier, A.: Adapting arbitrary normal mutation distributions in evolution strategies: The covariance matrix adaptation. In: Proc. 1996 IEEE Inter. Conf. on Evol. Comp., pp. 312–317 (1996)
14. Mika, S., Scholkopf, B., Smola, A., Muller, K.-R., Scholz, M., Ratsch, G.: Kernel PCA and De-Noising in Feature Spaces. *Advances in Neural Info. Processing Sys.* 11, 536–542 (1999)
15. Andoni, A., Indyk, P.: Near-Optimal Hashing Algorithms for Approximate Nearest Neighbor in High Dimensions. *Comm. of the ACM* 51(1), 117–122 (2008)
16. Hugues, L., Bredeche, N.: Simbad: an Autonomous Robot Simulation Package for Education and Research. In: Proc. Sim. of Adaptive Behavior (SAB 2006), pp. 831–842 (2006)

A Real-Time Personal Authentication System with Selective Attention and Incremental Learning Mechanism in Feature Extraction and Classifier

Young-Min Jang¹, Seiichi Ozawa², and Minho Lee¹

¹ School of Electrical Engineering and Computer Science, Kyungpook National University
1370 Sankyuk-Dong, Puk-Gu, Taegu 702-701, Korea

² Graduate School of Engineering, Kobe University
1-1 Rokko-dai, Nada, Kobe 657-8501, Japan

ymjang@ee.knu.ac.kr, ozawasei@kobe-u.ac.jp, mhlee@knu.ac.kr

Abstract. We propose a new approach for a real-time personal authentication system, which consists of a selective face attention model, incremental feature extraction, and an incremental neural classifier model with long-term memory. In this paper, a face-color preferable selective attention combined with the Adaboost algorithm is used to detect human faces, and incremental principal component analysis (IPCA) and resource allocating network with long-term memory (RAN-LTM) are effectively combined to implement real-time personal authentication systems. The biologically motivated face-color preferable selective attention model localizes face candidate regions in a natural scene, and then the Adaboost based face detection process identifies human faces from the localized face-candidate regions. IPCA updates an eigen-space incrementally by rotating eigen-axes and adaptively increasing the eigen-space dimensions. The features extracted by projecting inputs to the eigen-space are given to RAN-LTM which learns facial features incrementally without unexpected forgetting and recognizes faces in real time. The experimental results show that the proposed model successfully recognizes 200 human faces through incremental learning without serious forgetting.

Keywords: person authentication, face detection, selective attention, saliency map, incremental learning, principal component analysis, RBF networks.

1 Introduction

Recently, biometrics features have been broadly used as a means to authenticate user's identity. There have been considered various biometrics features to represent user's characteristics such as fingerprints, iris patterns, facial features, hand silhouettes which have their own merits and demerits for real world applications. Among authentication schemes using facial biometric features, the eigen-face approach, in which eigenvectors are computed to transform face image data into low-dimensional features, are widely adopted for face recognition systems. The biometrics

using facial information is one of the promising approaches to implementing a reliable system for personal authentication.

However, one of the difficulties to implement a facial feature based authentication system is to enhance the robustness over the spatial and temporal variations of human faces due to the growth (or aging) and the changes in lighting conditions, face directions, expressions, make-up, and so forth. Conventional personal authentication systems can achieve excellent performance when the system is tested over a benchmark dataset. However, it could drop rather drastically when they are operated in a practical environment. This is because the training set of face images will be either insufficient or inappropriate for future events.

Even if a large amount of face images are available during the construction of a personal authentication system, it is unlikely that all the variations that will happen in future could be considered in advance; thus reliable performance of the authentication system in practical situations can hardly be expected with only a static dataset. In this paper, as a solution for this problem, we propose a new personal authentication system that can learn continuously to adapt to incoming new training human faces. This can be done by embedding an incremental learning ability for both the feature extraction part and the classification part.

This paper is organized as follows; Section 2 describes the proposed incremental personal authentication system which consists of the bottom-up face detection using face color preferable attention for selecting face candidate areas [1], the incremental learning of the feature extraction part using incremental principal component analysis (IPCA) [2, 3], and the incremental learning of a neural classifier called resource allocating network with long-term memory (RAN-LTM) [3]. The experimental results will be followed in Section 3. Section 4 presents our conclusions and discussions.

2 Incremental Personal Authentication System

Figure 1 shows the proposed incremental personal authentication system. At first, we simply consider a skin color preferable attention model for face color perception and Haar-like form features for face form perception, in which all processes work in real time [1, 4].

A biologically motivated selective attention model with face-color preference can decide face candidate areas in a complex input scene. For the selected face candidate regions, an AdaBoost algorithm[4] using the Haar-like form feature is applied to selectively localize human faces not in all regions of the input scene but only in the face candidate areas obtained by the face color preferable selective attention model. Thus, we use a face candidate localizer based on the biologically motivated bottom-up saliency map (SM) model [5]. Second, we adopt IPCA for facial feature extraction conducted in an online way [2, 3]. Finally, we introduce a neural classifier called RAN-LTM which learns facial features incrementally without unexpected forgetting and recognizes faces using eigen-features obtained by IPCA [3]. The detail processing in each part is described below.

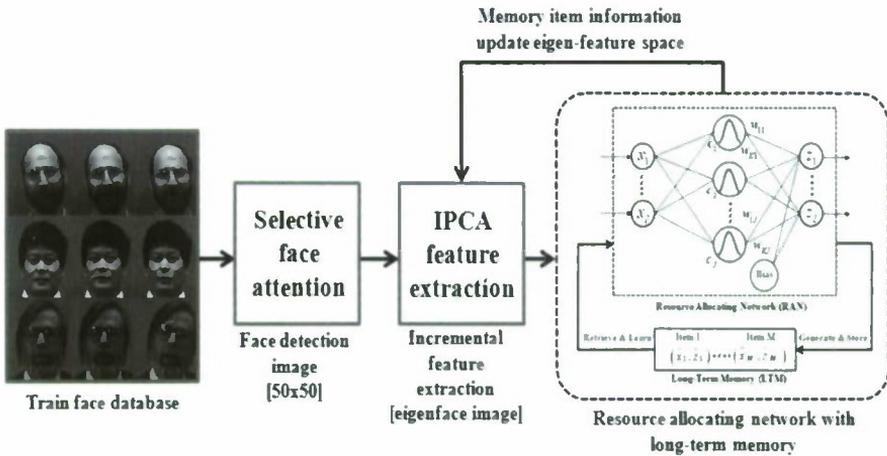


Fig. 1. The block diagram of processing in the incremental personal authentication system

2.1 Selective Attention Model with AdaBoost for Human Face Detection

In order to implement a human-like efficient visual selective attention function, we consider the bottom-up saliency map (SM) model proposed in [6]. The SM model reflects the functions of the retina cells, the lateral geniculate nucleus (LGN) and the visual cortex. Since the retina cells can extract edge and intensity information as well as color opponency, we use these factors as the basic features of the SM model [6-8]. In order to take the face color preference property into consideration, the skin color filtered [9] intensity feature is considered together with the original intensity feature. Depending on a given task to be conducted, those two intensity features are differently biased. For face preferable attention, a skin color filtered intensity feature works for a dominant feature in generating an intensity feature map. And the real color components red(R), green(G), blue(B), yellow(Y) are extracted using normalized color coding [7]. According to our experiments, the real color component R among 4 real color components shows dominant contribution for face color plausible filtering. Moreover, RG color opponent coding features also show a discriminate characteristic between face and non-face area. Therefore, in the proposed model, only the real color component R and RG color opponent features are considered to generate a skin color filter, which also plays a role for reducing computation time as well as getting better skin color filtering performance.

Actually, considering the function of the LGN and the ganglian cells, we implement the on-center and off-surround operation by the Gaussian pyramid images with different scales from 0 to *n*-th level, whereby each level is made by the sub-sampling of 2^n , thus it is able to construct four feature bases such as the intensity (I), and the edge (E), and color (RG and BY) [6, 8]. This reflects the non-uniform distribution of the retina-topic structure. Then, the center-surround mechanism is implemented in the model as the difference operation between the fine and coarse

scales of the Gaussian pyramid images [6, 8]. Consequently, three feature maps are obtained by the following equations.

$$\begin{aligned}
 I(c, s) &= | I(c) \ominus I(s) | \\
 E(c, s) &= | E(c) \ominus E(s) | \\
 RG(c, s) &= | R(c) \ominus G(c) | - | G(s) \ominus R(s) |
 \end{aligned} \tag{1}$$

where “ \ominus ” represents interpolation to the finer scale and point-by-point subtraction, c and s are indexes of the finer scale and the coarse scale, respectively. Features are combined into three feature maps as shown in Eq. (2) where \bar{I} , \bar{E} and \bar{C} stand for intensity, edge, and color feature maps, respectively. These are obtained through across-scale addition “ \oplus ” [6].

$$\begin{aligned}
 \bar{I} &= \bigoplus_{c=2}^3 \bigoplus_{s=c+2}^{c+3} N(I(c, s)) \\
 \bar{E} &= \bigoplus_{c=2}^3 \bigoplus_{s=c+2}^{c+3} N(E(c, s)) \\
 \bar{C} &= \bigoplus_{c=2}^3 \bigoplus_{s=c+2}^{c+3} N(RG(c, s))
 \end{aligned} \tag{2}$$

Thus, the three features maps such as \bar{I} , \bar{E} and \bar{C} can be obtained by the center-surround difference and normalization (CSD&N) algorithm [6]. A SM is generated by the summation of these three feature maps.

The salient areas are obtained by selecting areas with relatively higher saliency in the SM. In order to decide salient area, the proposed model generates binary data for each selected face candidate area using Otsu’s threshold method [10] in the SM. Then, the proposed model makes a group of segmented areas using a labeling method

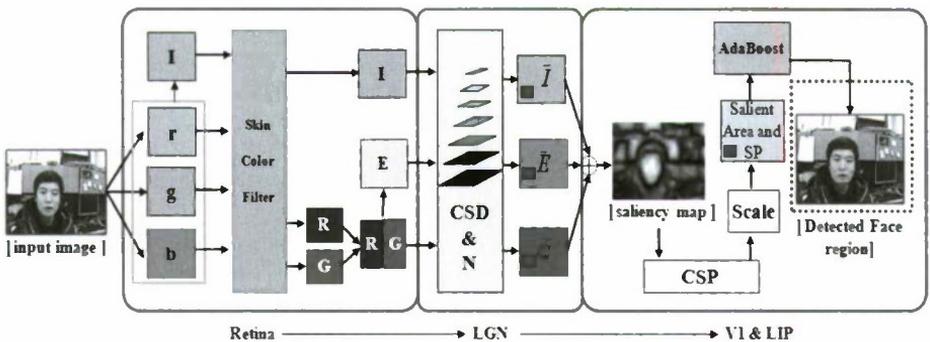


Fig. 2. The proposed selective attention model for human face detection: r: red, g: green, b: blue, R: real red, G: real green, I: intensity, E: edge, RG: red-green opponent coding, CSD&N: center surround difference & normalization, \bar{I} : intensity feature map, \bar{E} : edge feature map, \bar{C} : color feature map, SM: saliency map

for each binary face candidate area. After obtaining the candidate salient areas for human face, the obtained face candidate areas are used as input of the AdaBoost algorithm [4]. We adopted an AdaBoost approach using simple Haar-like features as the face detection algorithm for correctly localizing faces in the face candidate regions selectively selected by the face-color preferable SM model [1]. There are two data sets for face feature extraction and learning for the AdaBoost model. One is called a positive dataset in which every image has a face.

The other for non-face images set is called a negative dataset. For two data sets, Haar-like features are extracted in order to select the proper features and train the AdaBoost face detection model. The figure 2 shows the proposed selective attention model for human face detection.

2.2 Incremental Learning of Feature Extraction Using IPCA

In the IPCA [2], an eigen-feature space is updated through two operations: the rotation of eigen-axes and the dimensional augmentation. Assume that N training samples $x_i \in R^n$ ($i=1, \dots, N$) have been presented so far, and an eigenspace model $\Omega = (\bar{x}, U, A, N)$, is constructed by calculating the eigenvectors and eigenvalues from the covariance matrix of x_i , where \bar{x} is a mean input vector, U is an $n \times l$ matrix whose column vectors correspond to the eigenvectors, and A is an $l \times l$ matrix whose diagonal elements correspond to the eigenvalues. Here, l is the number of dimensions of the current eigenspace. Let us consider the case that the $(N+1)$ th training sample y is presented. The addition of y will lead to the changes in both of the mean vector and covariance matrix; therefore, the eigenvectors and eigenvalues should also be recalculated. The mean input vector \bar{x} is easily updated as follows:

$$\bar{x}' = \frac{1}{(N+1)}(N\bar{x} + y). \quad (3)$$

The problem is how to update the eigenvectors and eigenvalues. When the eigenspace model Ω is reconstructed to adapt to a new sample, we must check whether the dimensions of the eigenspace should change or not. If the new sample has almost all energy in the current eigenspace, the dimensional augmentation is not needed in reconstructing the eigenspace. However, if it has some energy in the complementary space to the current eigenspace, the dimensional augmentation cannot be avoided. This can be checked by the accumulation ratio whose incremental representation is given as follows:

$$A(l) = \frac{N(N+1) \sum_{i=1}^l \lambda_i + N \|U^T(y - \bar{x})\|^2}{N(N+1) \sum_{i=1}^n \lambda_i + N \|y - \bar{x}\|^2}. \quad (4)$$

If $A(l)$ is smaller than a threshold value θ , a new eigen-axis is added to the current eigenspace along the residue vector h :

$$h = (y - \bar{x}) - Ug \tag{5}$$

Where

$$g = U^T (y - \bar{x}). \tag{6}$$

It has been shown that the eigenvectors and eigenvalues can be updated based on the solution of the following intermediate eigenproblem [11]:

$$\left(\frac{N}{(N+1)} \begin{bmatrix} A & 0 \\ 0^T & 0 \end{bmatrix} + \frac{N}{(N+1)^2} \begin{bmatrix} gg^T & \gamma g \\ \gamma g^T & \gamma^2 \end{bmatrix} \right) R = RA' \tag{7}$$

where $\gamma = \tilde{h}^T (y - \bar{x})$, R is an $(l+1) \times (l+1)$ matrix whose column vectors are the eigenvectors obtained from the above intermediate eigenproblem, A' is the new eigenvalue matrix, and 0 is an l -dimensional zero vector. Using R , we can obtain the new $n \times (l+1)$ eigenvector matrix U' as follows:

$$U' = [U, \hat{h}] R \tag{8}$$

where

$$\hat{h} = \begin{cases} h / \|h\| & \text{if } A(l) < \theta \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

Here, θ is a threshold value. Figure 3 shows a general flow in the incremental feature extraction using IPCA.

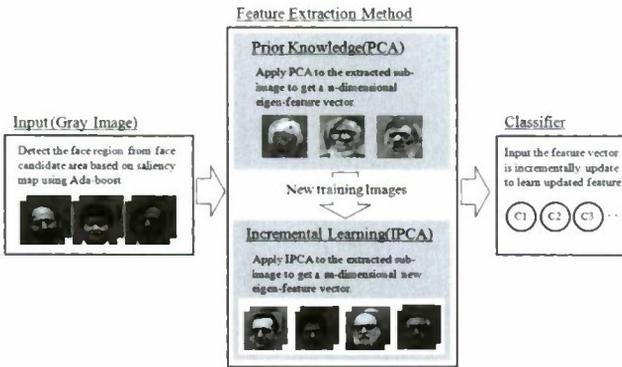


Fig. 3. The incremental feature extraction model: IPCA processing

2.3 Resource Allocating Network with Long-Term Memory

When training samples are incrementally given, neural networks often suffer from a well-known phenomenon called catastrophic interference [12]. RAN-LTM can alleviate this problem. Figure 4 shows the architecture of RAN-LTM which consists

of two parts: Resource Allocating Network (RAN) [13] and Long-Term Memory (LTM). RAN is an extended model of a Radial Basis Function (RBF) network in which the allocation of hidden units is automatically carried out. Let us denote the number of input units, hidden units, and output units as I, J, K , respectively. Moreover, let the inputs be $x = \{x_1, \dots, x_I\}^T$, the outputs of hidden units be $y = \{y_1, \dots, y_J\}^T$, and the outputs be $z = \{z_1, \dots, z_K\}^T$. The calculation in the forward direction is given as follows:

$$y_j = \exp\left(-\frac{\|x - c_j\|^2}{\sigma_j^2}\right) \quad (j = 1, \dots, J) \tag{10}$$

$$z_k = \sum_{j=1}^J w_{kj} y_j + \xi_k \quad (k = 1, \dots, K) \tag{11}$$

where $c_j = \{c_{j1}, \dots, c_{jI}\}^T$ and σ_j^2 are the center and variance of the j th hidden unit, w_{kj} is the connection weight from the j th hidden unit to the k th output unit, and ξ_k is the bias of the k th output unit. The items stored in LTM are called ‘memory items’ that correspond to representative input–output pairs. These pairs can be selected from training samples, and they are learned with newly given training data to suppress forgetting. In the learning algorithm, a memory item is created when a hidden unit is allocated: that is, an RBF center and the corresponding output are stored as a memory item in the LTM. The learning algorithm of RAN-LTM is divided into two phases: the allocation of hidden units (i.e. incremental selection of RBF centers) and the calculation of connection weights between hidden and output units. The procedure in the former phase is the same as that in the original RAN, except that memory items are created at the same time. Once hidden units are allocated, the centers are fixed afterwards. Therefore, the connection weights $W = \{w_{kj}\}$ are only parameters that are updated based on the output errors. To minimize the errors based on the least squares method, it is well known that the following linear equalities should be solved [14]:

$$\Phi W = D \tag{12}$$

where D is the matrix whose column vectors correspond to the target outputs. Suppose that a training sample (x, d) is given and M memory items $(\tilde{x}_m, \tilde{z}_m)$ ($m = 1, \dots, M$) have already been created, then the target matrix D are formed as follows: $D = \{d, \tilde{z}_1, \dots, \tilde{z}_m\}^T$. Furthermore, $\Phi = \{\varphi_i\}$ ($i = 1, \dots, M + 1$) is calculated from the training sample and memory items as follows:

$$\varphi_{i,j} = \exp\left(-\frac{\|x - c_j\|^2}{\sigma_j^2}\right), \quad \varphi_{i+1,j} = \exp\left(-\frac{\|\tilde{x}_i - c_j\|^2}{\sigma_j^2}\right) \quad (j = 1, \dots, J; \quad i = 1, \dots, M). \tag{13}$$

To solve W in Eq. (13), Singular Value Decomposition (SVD) can be used.

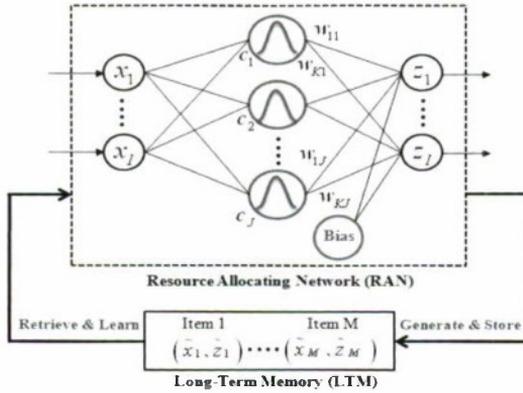


Fig. 4. The architecture of RAN-LTM

3 Experimental Results

Figure 5 shows a simulation process of the face detection model. Only the AdaBoost algorithm based on Haar-like form features generates some wrong face detection results. Fig. 5 (a) shows an example with a wrong face detection case which caused by considering Haar-like form feature only in an intensity image by the AdaBoost algorithm. In this case, a shirt is wrongly detected as a face since the intensity distribution in a shirt looks like a face. The problems can be resolved by the proposed model using face candidate areas as shown in Fig. 5 (b). A shirt is not selected as a face candidate area by the proposed face-color preferable attention model as shown in Fig. 5 (c), which is obtained from the face-color preferable attention model.

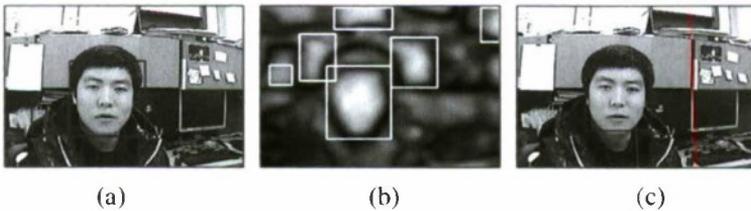


Fig. 5. Comparison of face detection between an AdaBoost algorithm and the proposed model; (a) face detection result by the AdaBoost, (b) face color preferable SM and face candidate area, (c) face detection result by the proposed model

A main goal of the proposed model is to reduce the time for face detection by restricting the searching regions using the selective attention model before conducting face detection by the AdaBoost. As shown in Table 1, the proposed model can successfully find human faces within 0.0539~0.2624 sec. The experiments were conducted for 530 facial images of the UCD database obtained in indoor environments [15]. In this experiment, we utilized the computer system with 3.0GHz CPU and 2Gbyte RAM.

Table 1. The time for face detection, and the performance comparison between the proposed model and Adaboost

		Adaboost	Proposed Model
Processing Time [ms]	Saliency Map	None	35.7 ms ~ 60.8 ms
	Adaboost	199.8 ms ~ 263.9 ms	7.75 ms ~ 240.1 ms
	Total	206.4 ms ~ 270.8 ms	53.9 ms ~ 262.4 ms
Performance (%)	True Positive	100%	100%
	False Positive	8.4%	3%

Figure 6 demonstrates how the incremental feature extraction using IPCA is conducted. Fig. 6 (a) shows an initial set of nine input faces, each of which is given by a gray-scale image. Fig. 6 (b) shows six eigen-faces (eigenvectors) computed by applying PCA to the initial set in a batch learning mode. Since an eigen-feature vector is obtained by projecting each face image to the six eigen-faces in Fig. 6 (b), every high-dimensional input image in Fig. 6 (a) is reduced to a six-dimensional eigen-feature vector. Fig. 6 (c) shows three sets of incrementally given data, each of which consists of two face images. After applying IPCA to these sets of face images, the number of eigen-faces is increased to 10 (i.e., 10-dimensional eigen-features are extracted) and the eigen-faces are updated as shown in Fig. 6 (d).

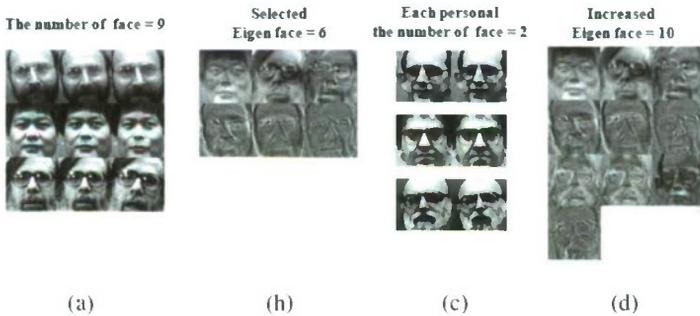


Fig. 6. A schematic processing flow in the incremental feature extraction by IPCA; (a) nine gray-scale input images, (b) six eigen-faces (eigenvectors) computed by PCA, (c) three sets of two face images that are given incrementally, and (d) updated eigen-faces whose number is increased to 10 by applying IPCA to the three sets of face images.

Table 2. Performance comparison between the two incremental learning models for personal authentication systems

	Baseline Model (IPCA with NN classifier)	Proposed Model (IPCA with RAN-LTM)
# of total image	Prior Knowledge face image : 3	
	Incremental Learning face image : 197	
Success	68	182
Fail	132	18
Performance (%)	34 %	91 %

Table 2 shows the comparisons between a baseline model using IPCA with the nearest neighbor (NN) classifier and the proposed model using IPCA with RAN-LTM, in which the number of output units of RAN-LTM is 200. As shown in Table 2, the proposed model successfully works as an incremental personal authentication system without serious forgetting.

4 Conclusions

In this paper, we propose a new approach to construct an adaptive personal authentication system, in which the system includes a face selective attention, incremental feature extraction by IPCA and an incremental neural classifier called RAN-LTM. The face selective attention model not only successfully localizes the facial areas but also appropriately rejects non-face areas. The proposed model is based on the face color related features in order to generate face color preferable attention and the AdaBoost algorithm decides whether the attended region contains a face characteristic. To learn a feature space incrementally, we adopt IPCA in which the feature space is update not only by rotating existing eigen-axes but also by increasing the number (i.e., the eigen-space dimensions are increased) based on the accumulation ration. To adapt to the evolution of the feature space, an extended model of RAN-LTM is adopted as a classifier, and we used an efficient way to reconstruct RAN-LTM after updating the feature space. In the experiments, we verify that the proposed incremental learning scheme works quite well and the test performance of the classifier is improved continuously as the incremental learning stages proceed.

As further work, we are planning to develop an embedded system for personal authentication based on facial biometrics information, and we should test the developed system for larger facial databases. Moreover, we are considering more experiments for verifying the proposed model by comparing the performance of the proposed model with that of state-of-the-art models.

Acknowledgments. This research was supported by the Converging Research Center Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2009-0082262) (50%) and also the grant funded by the Korea government (MEST) (2009-0070465) (50%).

References

1. Kim, B., Ban, S.W., Lee, M.: Improving AdaBoost Based Face Detection Using Face-Color Preferable Selective Attention. In: Fyfe, C., Kim, D., Lee, S.-Y., Yin, H. (eds.) IDEAL 2008. LNCS, vol. 5326, pp. 88–95. Springer, Heidelberg (2008)
2. Ozawa, S., Pang, S., Kasabov, N.: Incremental learning of chunk data for on-line pattern classification systems. *IEEE Trans. on Neural Networks* 19(6), 1061–1074 (2008)
3. Ozawa, S., Toh, S.L., Abe, S., Pang, S., Kasabov, N.: Incremental learning of feature space and classifier for face recognition. *Neural Networks* 18, 575–584 (2005)
4. Viola, P., Jones, M.J.: Robust real-time face detection. *International Journal of Computer Vision* 57(2), 137–154 (2004)

5. Ban, S.W., Lee, M., Yang, H.S.: A face detection using biologically motivated bottom-up saliency map model and top-down perception model. *Neurocomputing* 56, 475–480 (2004)
6. Jeong, S., Ban, S.W., Lee, M.: Stereo saliency map considering affective factors and selective motion analysis in a dynamic environment. *Neural Networks* 21, 1420–1430 (2008)
7. Goldstein, E.B.: *Sensation and Perception*, 4th edn. An International Thomson Publishing Company, USA (1996)
8. Choi, S.B., Jung, B.S., Ban, S.W., Niitsuma, H., Lee, M.: Biologically motivated vergence control system using human-like selective attention model. *Neurocomputing* 69, 537–558 (2006)
9. Kovač, J., Peer, P., Solina, F.: Human skin colour clustering for face detection. In: *EUROCON*, vol. 2, pp. 144–148 (2003)
10. Otsu, N.: A threshold selection method from gray-level histogram. *IEEE Trans. System Man Cybernetics*, 62–66 (1979)
11. Hall, P., Martin, R.: Incremental eigenanalysis for classification. *Proc. British Machine Vision Conference* 1, 286–295 (1998)
12. Carpenter, G.A., Grossberg, S.: The, A. R. T. of adaptive pattern recognition by a self-organizing neural network. *IEEE Computer* 21(3), 77–88 (1988)
13. Platt, J.: A resource-allocating network for function interpolation. *Neural Computation* 3(2), 213–225 (1991)
14. Haykin, S.: *Neural networks-A comprehensive foundation*, 2nd edn. Prentice Hall, Englewood Cliffs (1999)
15. UCD Valid Database, <http://ee.ucd.ie/validdb/datasets.html>

An Efficient Face Recognition through Combining Local Features and Statistical Feature Extraction

Donghyun Kim and Hyeyoung Park*

School of Computer Science and Engineering
Kyungpook National University
Sangyuk-dong, Buk-gu, Daegu, 702-701, Korea
{newpolaris,hyepark}@knu.ac.kr
<http://bclab.knu.ac.kr>

Abstract. This paper proposes a hybrid method for face recognition using local features and statistical feature extraction methods. First, a dense set of local feature points are extracted in order to represent a facial image. Each local feature point is described by the keypoint descriptor defined by SIFT feature. Then, the statistical feature extraction methods, PCA and LDA, are applied to the set of local feature descriptors in order to find low dimensional features. With the obtained low dimensional feature vectors, we can conduct face recognition task efficiently using a simple classifier. Through computational experiments on benchmark data sets, we show that the proposed method is superior to the conventional PCA and LDA in the classification performance. In addition, we also show that the proposed method can achieve remarkable improvement in the processing time compared to the conventional keypoint matching methods proposed for local features.

Keywords: Face recognition, Local features, Global statistical features, SIFT, PCA, LDA.

1 Introduction

Face recognition has attracted significant attention [1] in recent years because of its wide applications. One of the most widely used methods for efficient representation of facial images is the statistical feature extraction such as PCA (principal component analysis) and LDA (linear discriminant analysis). Through analyzing distributional properties of a set of facial images, these methods can find low dimensional features which maximizes specific statistical criteria. Eigenface method [2], which is based on PCA, provides low-dimensional representation of facial images that minimizes the loss of information in the sense of squared error. Fisherface method [3], which is based on LDA, provides low-dimensional representation that maximizes discrepancies among different classes.

* Corresponding author.

Though these methods can give highly effective dimension reduction properties, there are still difficult problems that should be considered. The statistical Eigenface and Fisherface methods consider a facial image as a vector point in a high dimensional input space, and they focus on finding distributional structure of whole data set. Consequently, the conventional statistical methods lack of keeping local features which is useful for discriminating human faces. In addition, the facial images to which PCA and LDA have been applied are usually represented by just gray level intensity. However, human visual system is known to use more sophisticated local feature descriptors such as gradient and orientation of local edges, which may play important role in recognizing face.

To overcome these restrictions of conventional statistical methods, we try to utilize local features for representing facial images. There have been various studies on developing local feature descriptors which are robust to various image transformations such as illumination change, rotation, and scale. Using these local feature descriptors, we can expect robust properties to local changes of images such as occlusions. One of the most successful local features for image data is SIFT (scale invariant feature transform), which is developed by Lowe [4]. Using feature descriptors defined by gradient and orientation of local image patches, Lowe suggested a method for object detection through extracting a set of keypoints from each image and matching them from two images using some invariant properties of the local features under the typical transformation such as scale, rotation, and translation [4].

However, in the case of facial recognition, the original SIFT method does not show satisfiable performance. Due to a lack of textures in facial images, original SIFT cannot detect enough number of keypoints from a face and thus represents the whole face using very limited number of local features. Moreover, facial images from even single subject have diverse variations which cannot be explicitly defined using mathematical relationship as like rotation, scale and translation. This characteristics may also be a cause deteriorating the performance of original SIFT. In order to resolve these problems, a number of variations of original SIFT have been proposed. The GRID-SIFT method, which was studied by Biecego and Luo [5], divides facial images into a number of subregions so that keypoint matching can be done in the corresponding subregions. Since the variation of facial images does not include the translation of facial part, this grid makes the matching process more efficient. However, this is a rudimentary approach and does not give substantial solution to the typical variational properties of facial images. On the other hand, the dense SIFT method has been developed, which constructs a dense set of keypoints for an image by extracting local features from fixed locations of each image [6][7]. Though dense SIFT can resolve the problem of the lack of keypoints, it is very costly in matching process because of the extremely large number of keypoints with high dimensional descriptors.

In this paper, we propose a combination of the statistical feature extraction method and the local keypoints descriptors in order to compensate their weak points and to augment the recognition performance. Based on the dense SIFT method, we represent a facial image using a dense set of local keypoints. Then we

apply PCA and LDA to the high dimensional vector composed of the dense set of keypoints, so as to get a low dimensional feature vector which is statistically meaningful and efficient in matching calculation. By using local features instead of gray level intensity when applying PCA and LDA, we expect to get more useful information for face representation. Also, by applying statistical feature extraction to the set of local keypoints, we expect to get more efficient low dimensional features which can learn the statistical variations of facial images.

In the next section, the conventional studies on local features (SIFT and dense SIFT) for face recognition are briefly reviewed. In Section 3, the proposed method for combining local feature approach and statistical feature extraction is described. Some experimental results on benchmark facial data set are given Section in 4, and conclusions are made in Section 5.

2 Local Feature Extraction For Face Recognition

In this section, we describe the conventional local features, SIFT and its modifications for face recognition. There are three issues when we use local features for face recognition. First, we need to determine how to select interesting point (i.e. keypoint) from an image. Second, we need to define an appropriate descriptor for the selected keypoints so that it can represent robust local properties of given images. After every image is represented by the set of keypoint descriptors, we need to measure the similarity between two images. In the local approaches, the similarity is measured through matching each keypoint in one image with one in the other image. Once the similarity is measured, we can conduct classification process using simple classifiers such as K-nearest neighbor. In this section, we briefly explain these three issues on SIFT method and its variations.

2.1 Keypoints Selection

SIFT [4] uses scale-space Difference-Of-Gaussian (DOG) to detect keypoints in images. For an input image, $I(x, y)$, the scale space is defined as a function, $L(x, y, \sigma)$ is produced from the convolution of a variable-scale Gaussian, $G(x, y, \sigma)$ with the input image. The DOG function is defined as follows:

$$\begin{aligned} D(x, y, \sigma) &= (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma) \end{aligned} \quad (1)$$

where k represents multiplicative factor.

The local maxima and minima of $D(x, y, \sigma)$ are computed based on its eight neighbors in current image and nine neighbors in the scale above and below. From the obtained local maxima and minima, keypoints are selected based on the measures of their stability and the value of keypoint descriptors which will be described below.

In face recognition, a main drawback of the original SIFT-based keypoint selection is that only a few numbers of keypoints are extracted due to a lack of textures of facial image, which may cause low performance in face recognition.

Thus, instead of the original keypoint selection method proposed by Lowe [4], local feature descriptors are extracted at regular image grid points so as to give us a dense description of the image content. Such modification is usually called as dense SIFT [8][9]. The dense SIFT was first developed in Dalal and Triggs [6] for pedestrian detection. Dreuw [7] have proposed to use the dense SIFT features for face recognition with grid matching strategy. We will also use this approach in the proposed combining method.

2.2 Keypoint Descriptor

Each keypoint extracted by SIFT is represented as a descriptor that is a 128 dimensional vector which is computed as a set of orientation histograms in neighborhood of the keypoint location. Each orientation histogram has 8 main direction which contains the summarized contents over 4 by 4 subregions by accumulation of gradient magnitude on each point. The gradient magnitude $m(x, y)$ and orientation $\Theta(x, y)$ is computed in Gaussian smoothed image L , which has the closest scale σ from the keypoint scale. The explicit computation of the magnitude of gradient $m(x, y)$ and the orientation $\Theta(x, y)$ at point (x, y) can be given as

$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2}, \quad (2)$$

$$\Theta(x, y) = \tan^{-1} \left\{ \frac{L(x, y+1) - L(x, y-1)}{L(x+1, y) - L(x-1, y)} \right\}. \quad (3)$$

In the original SIFT for object recognition, the gradients are aligned to the main direction for obtaining a rotation invariant descriptors.

In order to apply SIFT to face recognition, some modifications in descriptors have been done [7]. The main idea of the modification is, if face detector can provide an rotation-free image, descriptors are no longer needed to be rotation invariant. Moreover, the rotation invariant descriptors may even lead to false matching correspondences. Under this consideration, Dreuw [7] proposed to use upright version of the SIFT descriptor for face recognition, in which gradients of descriptor are aligned to a fixed direction. The upright versions are faster to compute and can increase accuracy.

2.3 Keypoints Matching and Classification

In order to classify an image data, we need to measure the similarity between two images. To measure the similarity using the set of keypoints, we first have to match each keypoint in an image to one in the other image. When we use the original SIFT method for face recognition, all possible pair of keypoints are traveled to select a set of matching pairs with sufficiently similar descriptors. The similarity of two images are then calculated as the number of selected matching pairs. However, in case of facial images, we cannot expect satisfiable performance through this measure with just local matching. Sometimes, a pair of keypoints from obviously different facial area (for example, one from left eye and the other from upper lips) is selected as a matching pair. Since the number of keypoint in

a facial image is quite small as we mentioned before, these mismatching pairs often results in wrong classification results. To avoid this, GRID-SIFT method divides an image into a number of subregion, and matching is allowed when two keypoints are from the same subregion, which leads slightly better performance. In the case of the dense SIFT, the same matching method can be applied. Since each keypoint is obtained from a fixed location, the silly mismatching of keypoints in different locations can be avoided to some extent. However, traveling all possible pair of keypoints is very time consuming process. Though the GRID approach can also be applied to speed up the matching process [7], it still needs high computational cost compared to statistical approaches.

3 Combination of Local and Statistical Feature Extraction

In order to solve the problem of high computational cost of the dense SIFT and to utilize statistical information of training data set, we try to combine the local features with the statistical feature extraction methods. In this paper, we exploit two well known statistical methods: PCA and LDA.

3.1 Statistical Feature Extraction

PCA tries to find a subspace whose basis vectors correspond to the maximum-variance directions in the original space, so as to minimize information loss caused by dimension reduction in the sense of squared error. Let W represent transformation matrix that provides an optimal linear transformation from the original space onto a subspace [10]. The new feature vectors \mathbf{y}_i is defined as follows:

$$\mathbf{y}_i = W^T \mathbf{x}_i, \quad (4)$$

where $i = 1, \dots, N$, N is the number of data.

The columns of W are the eigenvectors \mathbf{e}_i obtained by solving eigenvalue decomposition

$$\lambda_i \mathbf{e}_i = \Sigma \mathbf{e}_i, \quad (5)$$

where Σ is the covariance matrix of train data, λ_i is the eigenvalue associated with eigenvector \mathbf{e}_i

While PCA is an unsupervised method, LDA utilizes class information to give maximum class discrepancy. LDA tries to find a subspace in which the ratio of the between-class scatter S_b to the within-class scatter S_w is maximized. When the within-class scatter matrix S_w and the between-class scatter S_b are given by

$$S_w = \sum_{j=1}^c \sum_{i=1}^{N_j} (\mathbf{x}_i^j - \mu_j)(\mathbf{x}_i^j - \mu_j)^T, \quad (6)$$

and

$$S_b = \sum_{j=1}^c (\mu_j - \mu)(\mu_j - \mu)^T, \quad (7)$$

the columns of W can be obtained as the eigenvectors of $S_w^{-1}S_b$. Here, \mathbf{x}_i^j is the i th sample of class j , μ_j is the mean of class j , c is the number of classes, and N_j the number of samples in class j .

These methods can find statistically meaningful low dimensional features through learning from given data set, which cannot be obtained by local approaches with no learning process. However, they are basically global approaches, which treats an image as a vector in input space, and the obtained features mainly represents global shapes of faces.

3.2 Proposed Combination Strategy

In this paper, we try to combine the two different approaches to face recognition: the local feature matching and the global statistical feature extraction. First, we represent an image using local features. By using local features, we can obtain more abundant information from an image than by using the simple gray level intensity. Then we apply statistical feature extraction method to the set of image data represented by local features. Through statistical analysis on data set, we can expect to obtain low dimensional features which can efficiently represent diverse variations in the given training images.

Figure 1 shows an illustrative comparison between the conventional local approach and the proposed method. In the case of local approach, the whole dense set of keypoint are directly used for measuring similarity between a test image and training images. It is obvious that the computational cost for recognizing a test image increases depending on the number of training data as well as on the number of keypoints. In the case of proposed method, we conduct PCA and LDA to extract low dimensional features from the high dimensional local features. Though we need additional learning process in order to find the transformation matrix W , the cost for recognizing a test image is much lower than that of local approach. In addition, once PCA has been done for training data set, we do not need to keep the dense set of keypoint, which also requires large storage resource. In addition to economy of the computational resource, we can also expect to get statistically meaningful features representing diverse variations through learning of training images.

In the followings, the detail steps of the proposed method are given.

1. Let train face images, I_1, I_2, \dots, I_N , where N is the number of training images.
2. Apply dense feature extraction $f(x)$ on each images to obtain the matrix of descriptors:

$$D_i = f(I_i) \quad (8)$$

where $i = 1, 2, \dots, N$, D is $d \times m$ matrix, d is the dimensionality of each descriptor, and m is the number of keypoints.

3. By vectorizing each matrix D_i ($i = 1, \dots, N$), obtain a set of dm -dimensional vectors, $X = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$.

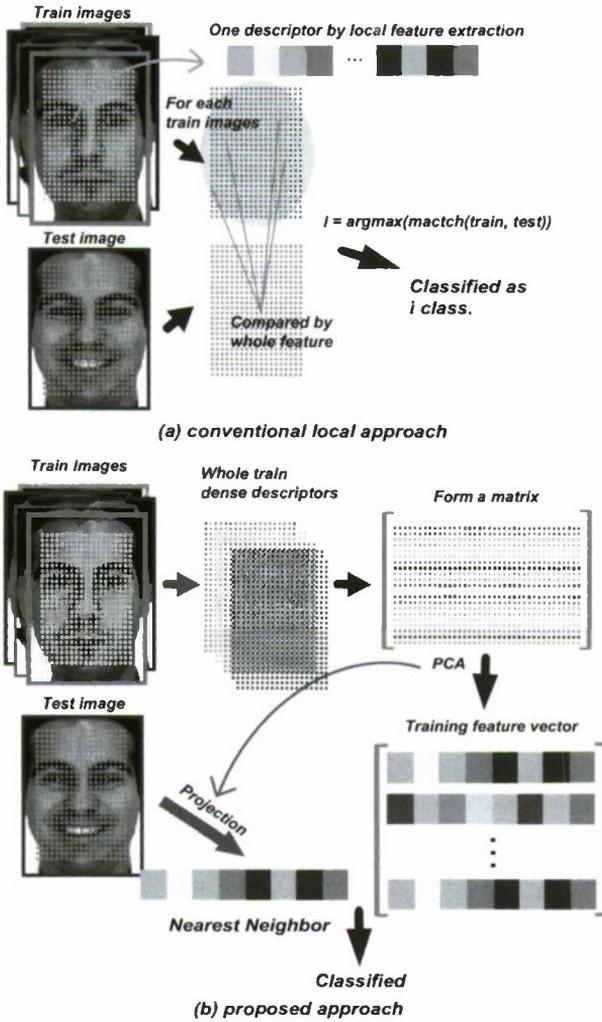


Fig. 1. An illustrative comparison between conventional local approach and proposed method

4. Apply PCA or LDA for X and get the linear transformation matrix W .
5. Transform each x_i using W to get low dimensional features y_i ($i = 1, 2, \dots, N$).
6. For a given test image, obtain low dimensional feature t through step 2,3, and 5.
7. Classify the test image via simple nearest neighbor algorithm.

4 Experimental Results

4.1 AR Database

In this section, we verify the efficiency of the proposed method through experiments on AR database [11], and provide comparisons with the conventional local approaches and the conventional statistical methods. The AR database consists of over 3,200 color images of frontal faces from 126 individuals: 70 men and 56 women. There are 26 different images for each person. For each subject, these were recorded in two different sessions separated by two weeks delay. Each session consists of 13 images which has differences in facial expression, illumination change and partial occlusion. In this experiment, we used manually aligned images [10] with the location of eyes. After localization, faces were morphed so as to fit a grid of size 85 by 60. Finally, images are resized to 88 by 64 pixels. A set of examples from one subject is shown in Fig. 2. The first and second row show images taken at first session, and the remaining images were taken at the second session.

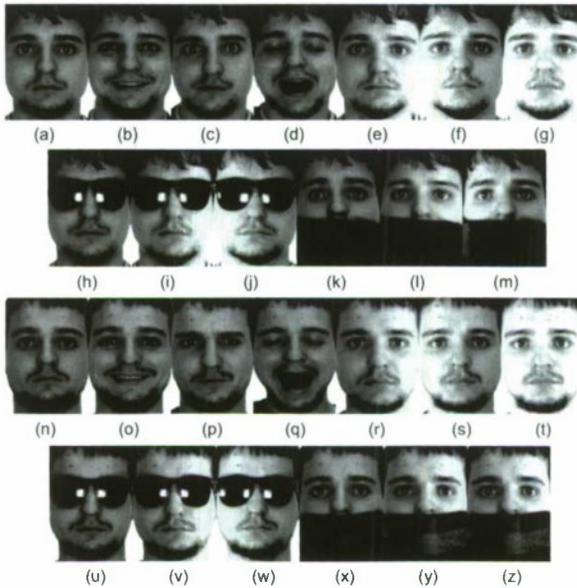


Fig. 2. Sample images for one subject of AR database

4.2 Experimental Conditions and Results

Using AR database, we compared the classification performance of the proposed method with a number of conventional methods: PCA, LDA, SIFT, and dense SIFT with variation in matching method. We used the open source implementation of SIFT and dense SIFT, which is implemented by Vedaldi and Fulkerson

[12]. For SIFT, we applied the original matching method proposed by Lowe [4], which was briefly described in Section 2.3. For dense SIFT (DSIFT), features are extracted at every two pixel points in row and column direction from each image. For keypoint matching strategy, we tried three variations. The basic DSIFT denotes the same matching strategy as that of the original SIFT. Since the original matching strategy tries to match all possible pairs of keypoints in an image, the computational cost becomes very high especially in the case of dense SIFT. The DSIFT 1-to-1 denotes matching a keypoint in an image to one at the same location of other image. Since DSIFT 1-to-1 matching has only one matching candidate for an image pair, the computational cost is much less than that for the original DSIFT. The DSIFT GRID denotes matching keypoints in the same sub-region. The DSIFT GRID, which has been proposed by Dreuw [7], can be considered as a compromise strategy between the above two methods. For PCA, we take the eigenvectors so that the loss of information is less than 1%, and discard first four eigenvectors, as usually done in application of PCA for face recognition. For LDA, we use the feature set obtained through PCA for avoiding small sample set problem. After applying LDA, we use maximum dimension of feature vector which is limited to the number of classes. For DSIFT PCA and DSIFT LDA, the same strategies as PCA and LDA are taken. As the comparison criteria, we used the mis-classification rates as well as the processing time. In order to show the relative time complexity among the methods, we showed the ratio of the processing time for each method to the time for PCA method (See Table 1 and 2).

In the first experiment, we used only non-occluded images with expression and illumination variations. For 100 individuals, seven non-occluded images taken at the first session (i.e., Fig. 2. (a)~(g)) were used for training, and the remaining seven non-occluded images from the second session (i.e., Fig. 2. (n)~(t)) were used for testing. The result of these experiments are listed in Table 1. Compared to the conventional PCA and LDA, we can see that the proposed method (DSIFT PCA and DSIFT LDA) achieves remarkable improvement in error rates. The original SIFT shows worst result as we can expect. Though DSIFT shows the best performances, the processing time for single testing is about 3600 times longer than the proposed method. The DSIFT GRID method can accelerate the speed, but still much slower than the proposed method. Compared to DSIFT 1-to-1 method that shows much shorter processing time than DSIFT GRID, we can see that the proposed method provides superior results. From this, we can say that the proposed method can achieve robustness to the variations in the training images to some extent.

In the second experiment, we compared the performance on the occluded images. For 100 individuals, three non-occluded images taken at the first session (i.e., Fig. 2. (a), (c), and (g)) were used for training, and four remaining non-occluded image and six occluded image from the first session (i.e., Fig. 2. (b), (d), (e), (f), (h)~(m)) were used for testing. The result of these experiments are listed in Table 2. We can see larger deterioration in the performance of PCA and LDA compared to the first experiment. This may be due to that the

global properties of the statistical method is not proper for the images with occlusion. Nevertheless, the proposed combination method achieves remarkable improvement by utilizing local features. Like the case of first experiment, DSIFT shows the best classification rates but the processing time for only single test is still terribly long. From these results, we can say that the proposed method is a reasonable compromise between classification rates and processing time.

Table 1. Result of face recognition on AR database with time delayed variation

strategy (options)		time (relative ratio)		Error Rate(%)
	number of features	single test	learning + test	
PCA	219	1.00	1.00	23.00
LDA	99	0.90	1.01	15.86
SIFT	depending on image	366.87	14.09	24.29
DSIFT	1120 × 128	532914.60	19290.77	0.14
DSIFT 1-to-1	1120 × 128	6473.95	239.88	9.29
DSIFT GRID [7]	1120 × 128	31317.92	1137.01	0.29
DSIFT PCA	568	148.23	14.32	2.14
DSIFT LDA	99	148.13	14.98	0.43

Table 2. Result of face recognition on AR database with occlusion

strategy (options)		time (relative ratio)		Error Rate(%)
	number of feature	single test	learning + test	
PCA	133	1.00	1.00	57.10
LDA	99	1.07	1.14	56.80
SIFT	depending on image	256.31	70.02	56.80
DSIFT	1120 × 128	255900.47	67333.72	0.00
DSIFT 1-to-1	1120 × 128	7708.85	2040.26	29.20
DSIFT GRID [7]	1120 × 128	20195.04	5325.11	0.00
DSIFT PCA*	252	223.31	77.24	5.00
DSIFT LDA*	99	223.26	77.62	3.90

5 Conclusions

In this paper, we proposed a hybrid approach to combine local features and statistical features. By using local features, we can get a robust representation for image data. By applying statistical feature extraction to the dense set of local features, we can find efficient low dimensional feature vectors. Since the utilization of local features and learning from data are two main ability of human being, which plays essential roles when human recognizes some objects, the proposed hybrid approach can be considered as a preliminary approach to realizing machines with more human-like visual pattern recognition ability. Throughout computational experiment, we showed that the proposed method maybe a reasonable compromise that keeping the both advantages of the local and statistical

features. This is a preliminary approach to combining local feature and global statistical approaches, and other sophisticated methods for extracting statistical features could be applied to get more improvement in classification performance.

Acknowledgements

This work was partially supported by National Research Foundation of Korea Grant funded by the Korean Government (2009-0082262).

This work was partially supported by KOSEF(Korea Science and Engineering Foundation) under Project Code R01-2007-000-20792-0.

References

1. Zhao, W., Chellappa, R., Phillips, P.J., Rosenfeld, A.: Face recognition: A literature survey. *ACM Comput. Surv.* 35(4), 399–458 (2003)
2. Turk, M., Pentland, A.: Eigenfaces for recognition. *Journal of Cognitive Neuroscience* 3(1), 71–86 (1991)
3. Belhumeur, P., Hespanha, J., Kriegman, D.: Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. In: Buxton, B.F., Cipolla, R. (eds.) *ECCV 1996*. LNCS, vol. 1065, pp. 43–58. Springer, Heidelberg (1996)
4. Lowe, D.G.: Distinctive image features from Scale-Invariant keypoints. *Int. J. Comput. Vision* 60(2), 91–110 (2004)
5. Bicego, M., Lagorio, A., Grosso, E., Tistarelli, M.: On the use of SIFT features for face authentication. In: *Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop*, p. 35. IEEE Computer Society, Los Alamitos (2006)
6. Dalai, N., Triggs, B., Rhone-Alps, I., Montbonnot, F.: Histograms of oriented gradients for human detection. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005*, vol. 1 (2005)
7. Dreuw, P., Steingrube, P., Hanselmann, H., Ney, H., Aachen, G.: SURF-Face: face recognition under viewpoint consistency constraints. In: *British Machine Vision Conference*, London, UK. (2009)
8. Bosch, A., Zisserman, A., Munoz, X.: Scene classification via pLSA. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) *ECCV 2006*. LNCS, vol. 3954, pp. 517–530. Springer, Heidelberg (2006)
9. Fulkerson, B., Vedaldi, A., Soatto, S.: Localizing objects with smart dictionaries. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) *ECCV 2008, Part I*. LNCS, vol. 5302, pp. 179–192. Springer, Heidelberg (2008)
10. Martinez, A.M., Kak, A.C.: PCA versus LDA. *IEEE Trans. Pattern Anal. Mach. Intell.* 23(2), 228–233 (2001)
11. Martinez, A., Benavente, R.: The AR face database. *CVC Technical Report #24* (June 1998)
12. Vedaldi, A., Fulkerson, B.: VLFeat: An open and portable library of computer vision algorithms (2008)

Parameter Learning in Bayesian Network Using Semantic Constraints of Conversational Feedback

Seung-Hyun Lee, Sungsoo Lim, and Sung-Bae Cho

Dept. Computer Science, Yonsei University

134 Sinchon-dong, Seodaemoon-ku

Seoul 120-749, Korea

{e2sh83,lss}@sclab.yonsei.ac.kr, sbcho@cs.yonsei.ac.kr

Abstract. Many learning techniques of Bayesian network have been developed for adaptation to user or environment. However, it seems several drawbacks still exists in conventional learning approach; the hardness of collecting log data, the inherent ambiguity in recognizing and reflecting implicit user's intention, and difficulties in extracting relations between data or definite rules. In this paper, we propose a method for parameter learning in Bayesian network using semantic constraints of conversational feedback to overcome these limitations. Production rules extracted from users' conversational feedback are used in parameter learning of Bayesian network. A comparison test with conventional approaches is conducted to verify the usefulness of the proposed method.

Keywords: Bayesian network, parameter learning, conversation, semantic constraints.

1 Introduction

Bayesian network (BN) is a graphical model to represent probabilistic relationships among a set of variables. The nodes represent variables in the DAG (directed acyclic graph) of BN and the directed arcs represent the relationship between variables. Over the last decade, BN has become a popular representation for encoding uncertain expert knowledge in expert systems [1].

Two methods are conventionally applied to determine the parameters of a BN: The use of expert knowledge and learning from data. Determining the parameters by the use of expert knowledge has the advantage of reflecting experts' preference, but the process is difficult and time-consuming. Furthermore, it is unclear whether the network designed by the experts is really the most appropriate model for the problem at hand. Therefore, there are studies which statistically learn parameters from training data [2–4]. If there are enough training data, we can get the proper probability of conditional probability table (CPT) using machine learning and statistical methods such as maximum likelihood estimation (MLE), sequential learning [5], EM algorithm [6], Gibbs sampling [7], and importance sampling [8].

However, the available data samples are often not enough when putting the learning theories into practice [9] and data distributions could be changed over time

according to the change of environment or users' preferences. Moreover, data samples from real world could have missing values and some other uncertainties. The conventional machine learning methods take much time to learn models from such data samples. Furthermore, they are hard to explicitly consider user's intention and are difficult in extracting symbolic relations between data or definite production rules.

In this paper, we propose a direct parameter learning method for Bayesian network based on semantic constraints extracted from conversation with users. Through conversation, the proposed method gets user's feedback and generates production rules in a symbolic representation. These rules are used to update parameters in Bayesian network in order to directly reflects the user's intention. Compared to the conventional data-driven learning methods, it can easily and instantly adapt to new environment and a user without a long period of observation. Furthermore, this approach gives user a chance to actively modify or develop its own probability network by just saying without prior knowledge on BN nor BN experts.

2 Related Works

2.1 Bayesian Networks

A Bayesian network has a shape of DAG (directed acyclic graph) expressing the relations of nodes and describes a large probabilistic relations with CPTs (conditional probability tables) constrained by the structure such as Fig. 1.

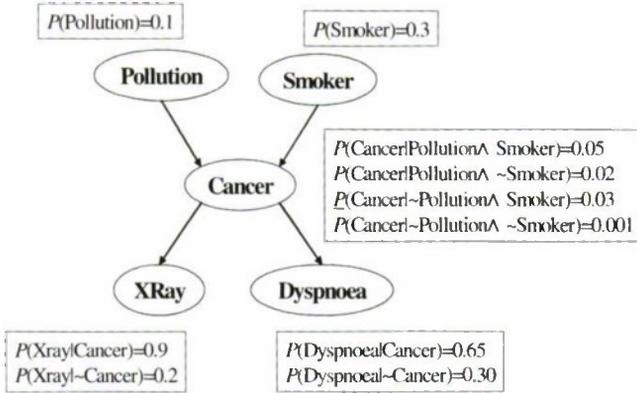


Fig. 1. An example of Bayesian network. The conditional probabilities are defined in the box.

Posterior belief of unobserved nodes, $Bel(h)$ is calculated by applying Bayes' Rule as follows:

$$Bel(h) = P(h|E) = \frac{P(E|h)P(h)}{P(E)} = \frac{P(h \wedge E)}{P(E)} \tag{1}$$

where h is the hypothesis of a node state and E represents a given evidence set E . The joint probability distribution is computed by Chain Rule as (2)

$$\begin{aligned}
 P(X_v) &= P(X_1, \dots, X_n) \\
 &= P(X_1)P(X_2 | X_1)P(X_3 | X_1, X_2) \dots \\
 &= \prod_{v \in V} P(X_v | X_{pa(v)})
 \end{aligned}
 \tag{2}$$

where $pa(v)$ denotes the set of parent variables of variable X_v for each node $v \in V$.

2.2 Conversational Interface

We utilize conversational agent described in [10] in order to enable conversational interaction with users for extraction of semantic rules. The conversational agent is composed of two parts: topic inference module using probabilistic network, and response selection module using keyword matching.

Topic Inference module: Overall user's intention which is implied in conversation is inferred based on Bayesian approach in this module. Context of possible conversation is modeled using BN which enables effective representation of relation between recognized token and its corresponding context. These relations are hierarchically captured into three levels: keyword, concept, and topic layer. The keyword layer consists of words related to topics in the domain. The concept level is composed of the entities or attributes of the domain, while the topic level represents entities whose attributes are defined.

Response Selection module: Proper pattern-response is selected by applying keyword matching technique according to the recognized current context in this module. Keyword matching is a procedure of searching the knowledge based associated with the topic of conversation. When there are many scripts, performance of keyword matching declines because of the time required to traverse massive information space. Conversational agent divides its knowledge base, scripts, into several concept so that it is able to keep scalability and portability which are important for flexible reaction to the various situation. This significantly reduces the number of scripts to be compared. Each script is stored in XML format. A set of candidate scripts are sequentially matched to find an appropriate response. The matching scores are calculated by the F-measure, which is a popular measurement in text classification. When there is a corresponding pattern-response pair, language generation is used to generate the answer.

3 Direct Parameter Learning Method of Bayesian Network

In this section, we describe how BN parameters can be directly learned from interaction with user. The description is divided into two parts: extracting production rules from conversation, and converting rules into BN parameters. The following section explains the detailed mechanism to achieve conversation-based learning.

3.1 Extract Production Rules from Conversation

A user queries conditions and desirable results to conversational agent when abnormal or unwanted services are provided or whenever user wants to teach its own system.

Conversational agent analyzes user's feedback and extracts semantic information for generating production rules from the conversation. However, it is not simple to control the semantic information in a form of natural language which may contain complex meaning. In this paper, symbol based representation is adapted to manipulate and maintain information and design a language model which is defined as a BNF grammar to produce the formal descriptions of symbols in any domain as shown in Table 1.

Table 1. BNF description of the proposed language framework

Non-terminal		Predicates
<Production-rule-description>	::=	IF <Pattern> THEN <Response>
<Pattern>	::=	<Symbol-sequence>*
<Symbol-sequence>	::=	<Single-symbol> not <Symbol-sequence>
		(<Sequential-symbols>)
		(<Simultaneous-symbols>)
		(<Domain-specific-symbols>)
<Sequential-symbols>	::=	<Single-symbol> then <Symbol-sequence>
<Simultaneous-symbols>	::=	<Single-symbol> and <Symbol-sequence>
		<Single-symbol> or <Symbol-sequence>
<Domain-specific-symbols>	::=	<Single-symbol>
		<Domain-specific-operator>
		<Symbol-sequence>
<Single-symbol>	::=	<Value> null
<Value>	::=	Symbol-name
		Domain-specific-characteristic
< Domain-specific-operator>	::=	Domain-specific-operator-name
<Response>	::=	<Single-symbol>*

The language model involves symbols and inferential rules which models the relations between symbols, and associates reasoning with the manipulation of the symbolic descriptions. A symbol has its own value, while the inferential rule is composed of the input patterns of symbols and the output symbol for replacement. The language basically describes the occurrence of symbols by means of concurrent and sequential relations [11].

A production rule is a sequence of one or more symbol sequences. A symbol sequence consists of sequential and simultaneous symbols. Sequential symbols occur one after the other in the order indicated by the sequence, while simultaneous symbols occur in parallel. The single symbol clause contains the basic information associated with the symbol extracted from conversation in a relevant domain. It consists of a unique symbol name and its various characteristics with respect to the application such as duration and intensity. In the language framework, the 'not' tag indicates that the symbol sequence should not include in the input. Especially, for the generality of the language framework, it allows to define a domain specific relation between symbols. New relationship such as relational reasoning and arithmetic operation can be defined according to domain.

Since the conversational information is semantically captured in the form of production rules, it requires a way to train the system using natural language. In our method, symbols are mapped to the corresponding words, while operators such as 'and' and 'or' are modeled with predefined templates. A query Q from the user is tokenized into a sequence of words $W = \{w_1, w_2, \dots, w_n\}$ by the lexical analysis. The pattern of W is analyzed by matching with predefined templates which is designed based on the language. We implement several functions that produce a part of production rules as follows.

$F_{rule}(\$symbol_1, \$symbol_2) \rightarrow \text{"IF } \$symbol_1 \text{ THEN } \$symbol_2\text{"}$

$F_{then}(\$symbol_1, \$symbol_2) \rightarrow \text{"} \$symbol_1 \text{ then } \$symbol_2\text{"}$

$F_{and}(\$symbol_1, \$symbol_2) \rightarrow \text{"} \$symbol_1 \text{ and } \$symbol_2\text{"}$

$F_{or}(\$symbol_1, \$symbol_2) \rightarrow \text{"} \$symbol_1 \text{ or } \$symbol_2\text{"}$

$F_{not}(\$symbol_1) \rightarrow \text{"not } \$symbol_1\text{"}$

$F_{specific}(\$symbol_1, \$symbol_2) \rightarrow \text{"} \$symbol_1 \text{ Domain-specific-operator } \$symbol_2\text{"}$

A number of templates are designed to implement a flexible dialogue for learning knowledge. Table 2 shows some examples of templates for knowledge learning. We can learn a new symbol based on these templates and find out the relations between each symbol.

Table 2. Templates defined for knowledge learning from conversation

Template1	IF $\$symbol_1$ 'is' $\$symbol_2$ 'and' $\$symbol_3$ THEN $F_{rule}(F_{and}(\$symbol_2, \$symbol_3), \$symbol_1)$ \rightarrow IF ($\$symbol_2$ and $\$symbol_3$) THEN $\$symbol_1$
Template2	IF $\$symbol_1$ 'is a sequence of' $\$symbol_2$ 'and' $\$symbol_3$ THEN $F_{rule}(F_{then}(\$symbol_2, \$symbol_3), \$symbol_1)$ \rightarrow IF ($\$symbol_2$ then $\$symbol_3$) THEN $\$symbol_1$
Template3	IF 'if' $\$symbol_2$ 'is occurred after' $\$symbol_3$ 'then' $\$symbol_1$ 'is activated' THEN $F_{rule}(F_{then}(\$symbol_2, \$symbol_3), \$symbol_1)$ \rightarrow IF ($\$symbol_3$ then $\$symbol_2$) THEN $\$symbol_1$
Template4	IF $\$symbol_1$ 'is true if' $\$symbol_2$ 'is false' THEN $F_{rule}(F_{not}(\$symbol_2), \$symbol_1)$ \rightarrow IF (not $\$symbol_2$) THEN $\$symbol_1$
Template5	IF $\$symbol_1$ 'is the sum of' $\$symbol_2$ 'and' $\$symbol_3$ THEN $F_{rule}(F_{specific}(\$symbol_2, \$symbol_3), \$symbol_1)$ \rightarrow IF ($\$symbol_2$ sum $\$symbol_3$) THEN $\$symbol_1$
Template6	IF 'if a person' $\$symbol_2$ 'then' $\$symbol_3$ ', she/he' $\$symbol_1$ THEN $F_{rule}(F_{then}(\$symbol_2, \$symbol_3), \$symbol_1)$ \rightarrow IF ($\$symbol_2$ then $\$symbol_3$) THEN $\$symbol_1$

3.2 Learning Parameters of Bayesian Network Based on Semantic Constraint

In order to keep the simplicity of learning problem, we restrict the problem space as below. First, every node in networks has maximum two states. Even if a node x has n ($n > 2$) states, we can split one node into n nodes with two states which enables or

disables the states in the original node x . Second, we assume that we already know the casual dependencies between variables in networks. All semantic relations extracted from conversation is already expressed as an arc between two variables in Bayesian network. For example, if we get semantic rule, $x \rightarrow y$, the structure, an arc from x to y is captured in the network.

The generated production rules (or semantic constraints) in Section 3.1 are used for direct learning parameters of Bayesian networks. The learning mechanism is composed of three steps. The first step is to find the Markov blanket M_x of a node x in a Bayesian network and the next step is to create a truth table about the nodes in $M_x \cup \{x\}$ based on semantic constraints. The last step of learning parameters is to create a conditional probability table (CPT) of the node x using the truth table created in the second step.



Fig. 2. A Markov blanket of a node A

In 1996, Koller and Sahami [12] proposed a cross-entropy based technique, known as Markov blanket for identifying redundant and irrelevant features. As shown in Fig. 2, the Markov blanket for a node x in a Bayesian network is the set of node M_x composed of x 's parents, its children, and its children's other parents. Formally, let B be a set of nodes which composes a Bayesian network and M_x be a subset of nodes which does not contain the node x , i.e., $M_x \cup B$ and $x \notin M_x$. M_x is a Markov blanket of the node x if x is conditionally independent of a distinct node y ($y \notin M_x$ and $y \neq x$) given M_x , i.e. $P(x | M_x, y) = P(x | M_x)$.

The Markov blanket of a node contains all the variables that shield the node from the rest of the network. This means that the Markov blanket of a node covers the range of knowledge needed to predict the behavior of the node. When parameters of node x is learned, therefore, we only limit the range to be updated as M_x instead of all the nodes in B .

After finding Markov blanket M_x corresponding node x , the suggested method generate a truth table depending on the values of node x and nodes in M_x . The truth table is used to figure out whether the production rules from Section 3.1 are fully satisfied. To generate the truth table, we regard each production rule as a proposition, and mark 'T' on the truth table if the proposition or its contraposition are satisfied, 'F' if the proposition is not satisfied, and 'X' the other cases. For example, the truth table, as shown in Table 3, can be constructed corresponding the production rule " $M_x \cup \{x\} = \{x, y\}$, if $x=1$ then $y=1$ ".

Table 3. Truth table

x	y	$x=1 \rightarrow y=1$
0	0	<i>T</i>
0	1	<i>X</i>
1	0	<i>F</i>
1	1	<i>T</i>

If $(x, y) = (1, 1)$, the production rule "if $x=1$ then $y=1$ " is satisfied so table value is '*T*'. If $(x, y) = (0, 0)$, the contraposition of the production rule "if $y=0$ then $x=0$ " is satisfied so table value is '*T*'. Also if $(x, y) = (1, 0)$, both the production rule and its contraposition are not satisfied so table value is '*F*'. If $(x, y) = (0, 1)$, both the production rule and its contraposition cannot be justified by the condition so table value is '*X*'.

Conversational agent can contains multiple production rules. When many production rules are given, the final truth table is generated as follows. Let C_i be the i th combination values of the truth table with $M_x \cup \{x\}$, the truth table value of C_i takes the value '*T*' if and only if there are one or more production rules that are satisfied the condition C_i and there is no production rules which is not satisfied the condition C_i . It takes the value '*F*' if and only if there are one or more production rules that are not satisfied the condition C_i . Otherwise it takes the value '*X*' which means every production rule cannot be justified on the condition C_i .

Table 4. Probability distribution table

x	y	Distribution
0	0	0.9
0	1	0.5
1	0	0.1
1	1	0.9

In order to assign specific values to the CPT of the node x , the data distribution or density table is needed rather than the truth table. Hence, we change the truth table to data distribution table by assigning specific values according to the value of the truth table, a is assigned to the value of '*T*', $(1 - a)$ to the value of '*F*' and 0.5 to the value of '*X*'. Using this data distribution table, finally, the value of CPT is determined. For instance, Table 4 shows the generated data distribution table using the result of Table 3 in a setting of a as 0.9. The probability $P(x=1|y=1)$ can be calculated as follows:

$$\begin{aligned}
 P(x=1|y=1) &= D(x=1, y=1) / (D(x=1, y=1) + D(x=0, y=1)) \\
 &= 0.9 / (0.9+0.5) \cup 0.64286
 \end{aligned}$$

where $D(x=a, y=b)$ denotes the value of the data distribution table.

4 Experimental Results

We evaluate the proposed conversation-based parameter learning algorithm (algorithm 1) with in smart home environment. In addition, the results are compared with two other algorithm, one is a conventional data-based learning method (algorithm 2) and another is a conversation-based approach with fixed learning weight for user input (algorithm 3), in order to validate our approach.

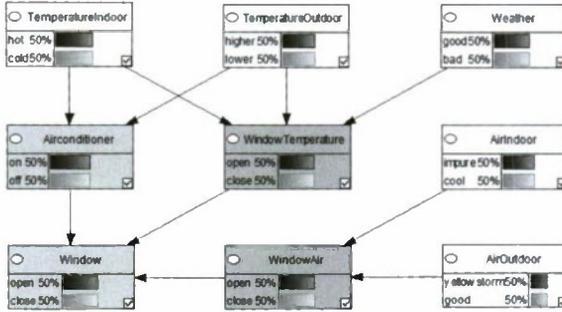


Fig. 3. Bayesian network of smart home agent designed for experiment

We design a BN module for smart home management as described in Fig. 3. It reflects the relationship between home appliance, home status, and outdoor environment whose aim is specifically for controlling air conditioner and window. The experiments were carried out by using dataset of possible situations stochastically generated according to our artificial home environment. Dataset has 2,000 situations of home status which half of them is used to learn parameters of BN and remaining data is for testing its accuracy. Accuracy of leaned model is evaluated by comparing dominant status according to the posterior belief with the solution contained in dataset.

Table 5. Conversational input from user in a form of natural language

Constraint	Dialogue
C1	"It's hot inside, and cool outside. Open up the window."
C2	"It's really hot today, and I can't stand hot like this. Turn on the air conditioner."
C3	"It's really cold today, I'm feeling cold inside. Why don't we close the window?"
C4	"It's freezing and I'm feeling cold. Close the window."
C5	"Air conditioner is on. Close the all windows."
C6	"It's raining hard. Close the window."

User input through conversation is applied for learning as described in Table 5. It shows the simple dialogues accordance with a possible situation which contains heuristic rules. These rules are learned by conversational agent mainly using the form of template 1 in Table 5. In this experiment, we assume all semantic rules are applied to BN model in a specific time rather than incremental application along the time line. For instance, parameters calculated from all conversational input are combined with parameter sets of BN after learning by 50, 100, 200, 300, and 400 data set, respectively in algorithm 1 and 3.

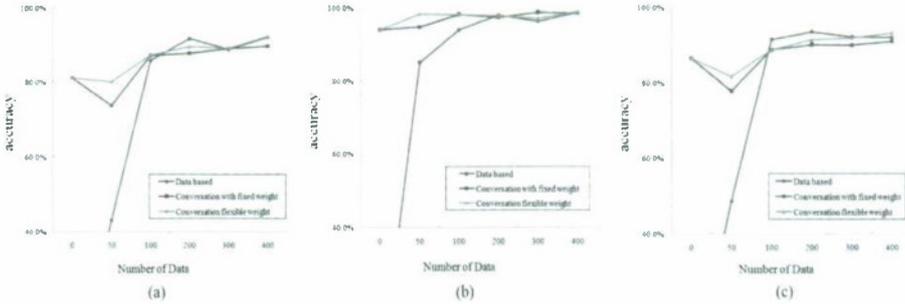


Fig. 4. Variations in accuracy for different level of observations

Table 6. Overall accuracy of three different learning method

Learning method	Window	Air conditioner	Overall
Data based learning	69.7%	78.7%	66.9%
Conversation & Data with fixed learning weight	87.4%	97.0%	84.7%
Conversation & Data with varying learning weight	88.9%	97.4%	86.5%

The results for three different learning algorithms are drawn in Fig. 4. (a), (b) and (c) are the result of inferred status of air conditioner, window and both, respectively. We can see all three algorithms successfully adapts the environment after enough data sets has been observed. However, there are huge gaps between algorithm 2 and algorithm 1 and 3 at the early stage. Due to the inherent feature of data-driven learning of BN, it is almost impossible to reflect environment exactly when data sets are small. Whereas, proposed conversational approach overcome this limitation. It shows definitely good performance is presented before observation of 100 data set because of its instant and direct learning from conversation. We confirm proposed method enables the system to capture features of given environment quickly with low effort without domain experts through the interaction with human.

Moreover, we can see algorithm 1 helps the system keep slightly more accuracy than others even when learning is mature. This means additive learning from conversation procedures can possibly leads more reliable agent system not only in

the initial phase but also in the stable phase. This is also supported by Table 6 which shows overall accuracy during the learning phase for each algorithm. Here, algorithm 3 has lower accuracy than algorithm 1. We can see it is important to control learning weight in the use of conversational based learning.

5 Conclusions and Future Works

In this paper, we proposed a direct parameter learning method for Bayesian network from the conversation with users. We defined functions and templates in order to extract semantic information from natural language and designed a language to facilitate a representation of relation between information in a symbolic description. We developed a novel learning method which includes mechanism of converting semantic rules into probability density for updating CPT. By applying this method, system can be easily adapted to new environment or a user without collecting much data and also keep high level of reliability. In addition, it gives user a chance to develop its own probability network that does not have any prior knowledge on BN. As the future work, we will extend the proposed method to structure learning of BN and apply to the case whose constraints are distributed in a time line.

References

1. Heckerman, D.: Bayesian networks for data mining. *Data Mining and Knowledge Discovery* 1, 79–119 (1997)
2. Cooper, H.G., Herskovitz, E.: A Bayesian method for the induction of probabilistic networks from data. *Machine Learning* 9, 309–347 (1992)
3. Buninc, W.L.: Operations for learning with graphical models. *Journal of Artificial Intelligence Research* 2, 159–225 (1994)
4. Heckerman, D., Geiger, D., Chickering, D.M.: Learning Bayesian networks: The combinations of knowledge and statistical data. *Machine Learning* 20, 197–243 (1995)
5. Cowell, R.G., Dawid, A.P., Sebastiani, P.: A comparison of sequential learning methods for incomplete data. *Bayesian Statistics* 5, 533–542 (1996)
6. Dempster, A., Laird, D., Rubin, D.: Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society* 39, 1–38 (1977)
7. Geman, S., Geman, D.: Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 721–741 (1984)
8. Riggelsen, G.: Learning parameters of Bayesian networks from incomplete data via importance sampling. *International Journal of Approximate Reasoning* 42, 69–83 (2006)
9. Feelders, A., van der Gaag, L.C.: Learning Bayesian network parameters under order constraints. *International Journal of Approximate Reasoning* 42, 37–53 (2006)
10. Hong, J.-H., Lim, S.-S., Cho, S.-B.: Autonomous language development using speech-act template and genetic programming for a conversational agent. *IEEE Trans. on Evolutionary Computation Special Issue on AMD*, 213–225 (2007)
11. Amir, E., Maynard-Zhang, P.: Logic-based subsumption architecture. *Artificial Intelligence* 153, 167–237 (2004)
12. Koller, D., Sahami, M.: Toward optimal feature selection. In: *13th International Conference on Machine Learning*, pp. 284–292 (1996)

Keystroke Dynamics Extraction by Independent Component Analysis and Bio-matrix for User Authentication

Thanh Tran Nguyen, Thai Hoang Le, and Bac Hoai Le

Abstract. Keystroke dynamics is unique specific characteristics used for user authentication problem. There are many researches to detect personal keystroke dynamics and authenticate user based on these characteristics. Most researches study on either the key press durations and multiple key latencies (typing time) or key-pressed forces (pressure-based typing) to find the owned personal motif (unique specific characteristic). This paper approaches to extract keystroke dynamics by using independent component analysis (ICA) through a standardized bio-matrix from typing sound signals which contain both typing time and typing force information. The ICA representation of keystroke dynamics is effective for authenticating user in our experiments. The experimental results show that the proposed keystroke dynamics extraction solution is feasible and reliable to solve user authentication problem with false acceptance rate (FAR) 4.12% and false rejection rate (FRR) 5.55%.

Keywords: Behavioural biometrics, keystroke dynamics, pattern recognition, independent component analysis (ICA), user authentication.

1 Introduction

Keystroke dynamics is detected from user characteristics based on how he types on the keyboard but not what he types. Keystroke dynamics captures typing characteristics such as key press duration 'dwell time' when typing and digraphs or serigraphs times - the latency between striking successive keys. All attributes of user extracted from typing are linked to user's profile through learning machine system. They are used to verify user by detecting his typing characteristics in the next time. In the previous report on keystroke dynamics, the characteristics are analyzed in novel concept with long text (see [1], [2]). Almost recent publication, keystroke dynamics can be retrieved in short text input concept like user ID and password. Various algorithms and methods are researched to apply for authenticating keystroke dynamics, such as: fuzzy algorithms [3], neural network - support vector machine [4] and multiple sequence alignment [5]. Besides the method based on typing time method, the pressure-based typing method is proposed ([6], [7]). All publications prove that keystroke dynamics can be used to improve security like physiological methodologies.

All above publications approach to detect keystroke dynamics based on either typing time or typing force. Our approach uses both characteristics to solve

the user authentication problem. In this paper, we propose an indirect method to detect key-pressed time, key-released time and key-typed forces by analyzing sound signals created when typing on keyboard. Fig. 1 summarizes our proposed user authentication method's process. Keystroke dynamics characteristics are retrieved from sound signals by using a sound recorder. In pre-processing phase, typing sound signals containing both characteristics are standardized and translated to a keystroke dynamics bio-matrix. The keystroke dynamics bio-matrix represents the unique characteristics of user's typing habit. To extract keystroke dynamics feature, independent component analysis (ICA) method is applied. ICA is a recently developed method in which the goal is to find a linear representation of non-Gaussian data so that the components are statistically independent, or as independent as possible. Such a representation seems to capture the essential structure of the data in many applications, including feature extraction and signal separation [8]. Face recognition [9] and facial feature extraction [10] are examples using ICA. In this paper, we use the ICA second architecture described in [9] to extract keystroke dynamics features from the bio-matrix. Experimental results show that our approach using the keystroke dynamics bio-matrix, ICA extraction method and neural network (Fast Artificial Neural Network Library - FANN [16]) for recognition is feasible and reliable to solve user authentication problem with 4.12% FRR and 5.55% FAR.

The remainder of this paper is organized as follows. In section 2, we present about an overview of the solution, pre-processing phase and the keystroke dynamics bio-matrix. Section 3 describes the architecture II to apply ICA method for extracting keystroke dynamics from the bio-matrix. Experimental results of the solution combining bio-matrix, ICA extraction method and FANN are reported in section 4. Conclusion and future works will be mentioned in section 5.

2 Keystroke Dynamics Represented by Bio-matrix

2.1 Indirect Method to Measure Keystroke Dynamics

The process illustrated in Fig. 1 has two phases: registration and authentication. In registration phase, user is required to input his username and password N_R times (15 times in our experiments). Of N_R register times, there are N_{RS} times in silent environment without noise to determine initial parameter values. After registering, user will be authenticated when accessing the system again. The sound signals received when user types on keyboard are analyzed. The spectro sound signals of typing pattern are standardized and translated to the keystroke dynamics bio-matrix in pre-processing phase. The ICA second architecture, then, is applied to extract keystroke dynamics features from the bio-matrix. The feature vector (ICA representation) is used as an input of FANN for training (registration) and testing (authentication).

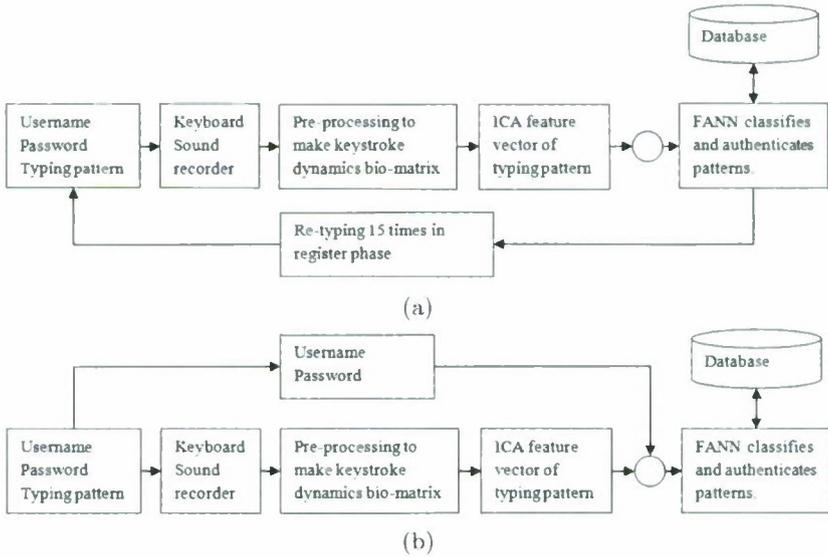


Fig. 1. Registration (a) and authentication (b) using keystroke dynamics

2.2 Pre-processing

The original sound signal is pre-processed to make the correlative keystroke dynamics bio-matrix. Fig. 2(a) is an example of sound signals of pressing and releasing keys. It also shows the difference in typing forces. The sound signal is transformed to frequency domain by short-time Fourier transform. Gabor transform is used to analyze typing sound signal because this transformation has no cross-term and avoids the confusion between noise and non-noise components. Moreover, this transformation has lower computational complexity so it improves the speed. Fig. 2(b) displays spectrogram of 'onetntall' typing pattern.

At registration phase, with the first N_{RS} registering times in silent environment, the threshold values are calculated for each user (including high frequency threshold θF_{high} and low frequency threshold θF_{low}).

$$\theta F_{high} = \frac{\sum_{i=1}^{N_{RS}} \max(f_j^i)}{N_{RS}} \tag{1}$$

$$\theta F_{low} = \frac{\sum_{i=1}^{N_{RS}} \min(f_j^i)}{N_{RS}} \tag{2}$$

where, N_{RS} is number of register times in silent environment, f_j^i is frequency value of the i^{th} time, j is index of signal frequency for each register.

The spectrogram of original sound signal is used to create the keystroke dynamics bio-matrix described in next section.

2.3 The Keystroke Dynamics Bio-matrix

The original typing signal is filtered by band-pass filter with θF_{high} , θF_{low} in order to get exact typing frequency domain. An intensity matrix $MI_{N_T \times N_F}$ is made from that domain which each element of the matrix is calculated in formula (5).

$$\delta_T = \frac{T}{N_T} \tag{3}$$

$$\delta_F = \frac{\theta F_{high} - \theta F_{low}}{N_F} \tag{4}$$

where, T is the time that user inputs password, N_T is predefined number of sections of T time, δ_T is time length of each time section; N_F is predefined number of divided sections in $[\theta F_{low}, \theta F_{high}]$ interval, δ_F is length of each frequency section.

$$MI_{x,y} = \sum_{i=(x-1)\delta_T}^{x\delta_T} \sum_{j=(y-1)\delta_F + \theta F_{low}}^{y\delta_F + \theta F_{low}} I_{i,j} \tag{5}$$

where, $x = 1..N_T$, $y = 1..N_F$, $I_{i,j}$ is intensity of frequency f_j^i .

From the intensity matrix, maximum intensity and minimum intensity of all elements are calculated in formula (6), (7).

$$I_{max} = \max(MI_{x,y}) \tag{6}$$

$$I_{min} = \min(MI_{x,y}) \tag{7}$$

where, $x = 1..N_T$, $y = 1..N_F$.

We propose the keystroke dynamics bio-matrix $bioM_{N_T \times N_F}$ whose elements (called bio-cell) represent the correlative intensity of the elements of the intensity matrix $MI_{N_T \times N_F}$.

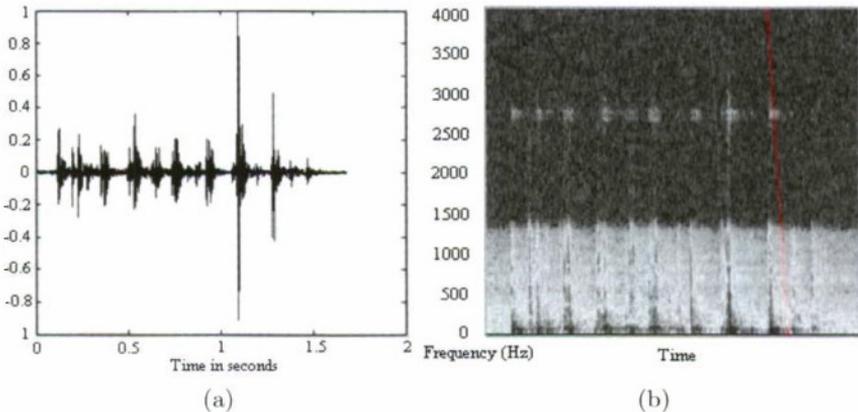


Fig. 2. (a) Time-sequence signal of password ‘onetntall’. (b) Spectrogram of password ‘onetntall’.

$$bioM_{x,y} = \left\lfloor \frac{MI_{x,y} - I_{min}}{I_{max} - I_{min}} \times N_I \right\rfloor + 1 \quad (8)$$

where, N_I is predefined number of intensity sections of the intensity matrix $MI_{N_T \times N_F}$.

In the next section, we describe the ICA second architecture to extract feature vector from the keystroke dynamics bio-matrix.

3 Extracting Keystroke Dynamics Feature by ICA

Independent Component Analysis (ICA) minimizes both second-order and higher-order dependencies in the input data and attempts to find the basis along which the data (when projected onto them) are - statistically independent. Bartlett *et al* [9] provided two architectures of ICA for face recognition task: Architecture I - statistically independent basis images, and Architecture II - statistically independent coefficients.

In this keystroke dynamics recognition problem, our goal is to find coefficients of feature vectors to achieve the most independent in desire. Therefore, in this paper, we selected architecture II of ICA method for the keystroke dynamics representation. A number of algorithms for performing ICA have been proposed (see [8] for reviews). In this paper, we apply FastICA algorithm developed by Aapo Hyvärinen [8] for our experiments.

Architecture II: Statistically Independent Coefficients. The goal in this approach is to find a set of statistically independent coefficients. A similar approach was used for face recognition [9] and for facial feature extraction [10].

We organize a data matrix X so that keystroke dynamics bio-matrices are in columns and the bio-cells are in rows. Bio-cell i and j are independent if when moving across the entire set of the bio-matrices, it is not possible to predict the value taken by bio-cell i based on the corresponding value taken by bio-cell j on the same bio-matrix. The goal in architecture I is using ICA to find a set of statistically independent basic bio-matrices. Although basic bio-matrices found in architecture I are approximately independent, when projecting down statistically independent basic bio-matrices subspace, feature vectors of each bio-matrix are not necessarily independent. Architecture II uses ICA to find a representation which coefficients are used to represent a bio-matrix in the basic bio-matrices subspace being statistically independent. Each row of weight matrix W is a bio-matrix. A , an inverse matrix of W , contains basic bio-matrices in its columns. Statistically independent coefficients in S will be recovered in columns of U (see Fig. 3); each column of U contains coefficients for combination of basic bio-matrices in A to construct bio-matrices of X .

Architecture II is implemented through the following steps:

Assumption that we have n bio-matrices; each bio-matrix has p bio-cells. Therefore, data matrix X has an order of $p \times n$.

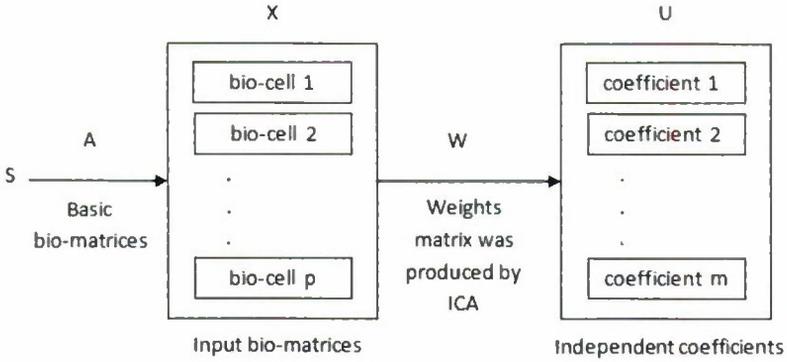


Fig. 3. Finding coefficients which presentation bio-matrices are independent

1. Let R be a $p \times m$ matrix containing the first m eigenvectors of a set of n keystroke dynamics bio-matrices in its columns.
2. Calculating set of principle component of set of bio-matrices in X :

$$C = R^T \times X \tag{9}$$

3. The coefficients for linearly combining the basic bio-matrices in A are determined:

$$U = W \times C \tag{10}$$

Assumption that we have a set of bio-matrices for testing X_{test} , feature extraction of X_{test} is computed through the following steps: firstly, from X_{test} , we calculate a set of principle component of X_{test} by:

$$C_{test} = R^T \times X_{test} \tag{11}$$

Then, a set of feature vectors of X_{test} in the basic bio-matrices space is calculated by:

$$U_{test} = W \times C_{test} \tag{12}$$

Each column of U_{test} is a feature vector corresponding with each bio-matrix of X_{test} .

Firstly, to keystroke dynamics representation with ICA method, we apply PCA to project the data into a m dimensional subspace with purpose to control the number of independent components made by ICA, and then ICA is applied to the eigenvectors to minimize the statistical dependence of feature vectors in the basic bio-matrices space. Thus, PCA uncorrelated input data, high-order dependence remain will be separated by ICA.

4 Experimental Results

In our experiments, N_U users are invited to test the proposed authentication system with 2 experiments. Experiment 1 is to authenticate in silent environment without any noise. Experiment 2 is to authenticate in both silent environment and workable environment (e.g. library, school yard, coffee shop ...).

Table 1. Parameters of experiments 1 and 2

Experiment	N_U	N_R	N_{RS}	N_{Auth}	N_{Attack}	N_T	N_F	N_I
1	20	15	15	10	10	100	100	256
2	20	15	5	10	10	100	100	256

In each experiment, after registering, user accesses the system N_{Auth} times to test authentication ability of the system. In addition, every user's username and password is public and five other persons will use that information to attack the system. An intruder will attack one account N_{Attack} times. We choose the number of time sections N_T is 100, the number of frequency sections N_F is 100 and the intensity sections N_I is 256. Table 1 shows the parameters of experiment 1 and experiment 2.

The recognizer was implemented by the neural network method. Fast Artificial Neural Network is used in our experiments. Fast Artificial Neural Network Library (FANN [16]), which is a free open source neural network library, implements multilayer artificial neural networks in C language and supports for both fully connected and sparsely connected networks. FANN has been used in many studies. FANN implementation includes:

Training (registration) step: assumption that we have N_U classes (N_U different users), training with FANN will create N_U sets of weights. Each set of weights corresponds with each class (each user).

Testing (authentication) step: the input is the ICA feature vector of user's keystroke dynamics bio-matrix (one of the N_U users mentioned above); this feature vector is tested with N_U sets of weights which were created in the training step, this user belongs to the class which corresponds with the set of weights having the biggest output.

One of our experimental results show that the proposed authentication system is acceptable with 4.2% FAR, 5.6% FRR in silent environment and 3.9% FAR, 6.1% FRR in workable environment. Table 2 shows the results of experiments 1 and 2. The results in both silent environment and workable environment are not deviated so much. It shows that the keystroke dynamics features extracted by ICA are quite different in silent environment or workable environment.

Table 2. Total FAR and FRR for experiments 1 and 2

Experiment	Number of authentic participants	Number of intruder participants	Number of attacks	Number of successful attacks	FAR%	FRR%
1	20	20	1000	42	4.2	5.6
2	20	20	1000	39	3.9	6.1

Table 3. Experimental results

Group	Silent environment		Workable environment	
	FAR%	FRR%	FAR%	FRR%
1	4.20	5.60	3.90	6.10
2	4.10	5.50	3.80	6.20
3	3.90	5.70	3.70	6.20
4	4.00	5.20	3.70	6.30
5	4.50	5.00	4.00	6.00
6	4.10	5.70	3.70	6.10
7	4.00	5.70	4.00	5.90
8	4.20	5.50	4.00	6.00
9	3.90	5.90	3.60	6.40
10	4.20	5.80	3.80	6.10
11	4.10	5.40	3.90	6.00
12	4.30	5.70	3.80	5.80
13	4.00	5.50	3.90	6.40
14	4.20	5.60	4.00	6.20
15	4.10	5.50	3.70	6.10
Average	4.12	5.55	3.83	6.12

Table 4. comparison of our results with previous efforts

Research	Number of participants	Training samples	Password string	FAR%	FRR%
Legget and Williams (1988) [11]	36	12	large	5.00	5.50
Joyce and Gupta [12]	33	8	4	13.30	0.17
De Ru and Eloff [13]	29	Varies (2 to 10)	1	2.80	7.40
Haider et al. [14]	Not mentioned	15	1	6.00	2.00
Araujo et al. [15]	30	10	1	1.89	1.45
Eltahir et al. [6]	23	20	1	3.75	3.04
Kenneth Revett [5] (threshold 0.60)	20	10	1	0.80	0.90
Our proposed method (silent environment)	20	15	1	4.12	5.55
Our proposed method (workable environment)	20	15	1	3.83	6.12

Other groups are invited to test the system like two above experiments. The results when testing in silent and workable environments are summarized in table 3. They show that the performance of proposed system is feasible and reliable.

Table 4 shows a comparison between results obtained here and previous research efforts. Note that these systems use different sample sizes with different parameters and methodologies to measure the results. Nevertheless, our proposed method gives comparable results with existing methods. This shows the feasibility and reliability of using sound signals to measure keystroke dynamics for authentication.

5 Conclusion

This study proposed the indirect method to measure the pressure of key typing via sound signals so widespread deployment is easier because it does not use any specific device like bio-keyboard. In addition, the novel keystroke dynamics bio-matrix combines both typing time and typing force. It is converted to ICA feature vector to authenticate user by FANN reliably. The experimental results show that the proposed authentication system is feasible and reliable. Besides that, it shows that the keystroke dynamics extraction using ICA second architecture is effective and stable in different environments. In future, we continue experiment with many groups of users in order to apply this authentication solution in practical problem having a lot of users.

References

1. Peacock, A., Ke, X., Wilkerson, M.: Typing patterns: A Key to User Identification. *IEEE Security and Privacy* 2(5), 40–47 (2004)
2. Curtin, M., Tappert, C.C., Villani, M., Ngo, G., Simone, J., Fort, H.S., Cha, S.: Keystroke Biometric Recognition on Long-Text Input: A Feasibility Study. In: *Proc. Int. Workshop Sci. Comp/Comp. Stat. (IWSCCS 2006)*, Hong Kong (June 2006)
3. Tapiador, M., Siguenza, J.A.: Fuzzy keystroke biometrics on web security. In: *AutoID 1999 Proceedings Workshop on Automatic Identification Advanced Technologies*, pp. 133–136. IEEE, Los Alamitos (1999)
4. de Oliveira, M.V.S., Kinto, E., Hernandez, E.D.M., de Carvalho, T.C.: User authentication based on human typing patterns with artificial neural networks and support vector machines. In: *SBC* (2005)
5. Revett, K.: A Bioinformatics Based Approach to User Authentication via Keystroke Dynamics. *International Journal of Control, Automation, and Systems* 7(1), 7–15 (2009)
6. Eltahir, W.E., Salami, M.J.E., Ismail, A.F., Lai, W.K.: Design and Evaluation of a Pressure-Based Typing Biometric Authentication System. *EURASIP Journal on Information Security*, Article ID 345047 2008, 14 (2008)
7. Eltahir, W.E., Salami, M.J.E., Ismail, A.F., Lai, W.K.: Dynamic Keystroke Analysis Using AR Model. In: *Proceedings of the IEEE International Conference on Industrial Technology (ICIT 2004)*, Hammamet, Tunisia, vol. 3, pp. 1555–1560 (2004)
8. Hyvärinen, A., Oja, E.: Independent Component Analysis: Algorithms and Applications. *Neural Networks* 13(4-5), 411–430 (2000)

9. Bartlett, M.S., Movellan, J.R., Sejnowski, T.J.: Face Recognition by Independent Component Analysis. *IEEE Transactions on Neural Networks* 13(6) (November 2002)
10. Do, T.T., Le, T.H.: Facial Feature Extraction Using Geometric Feature and Independent Component Analysis. In: Richards, D., Kang, B.-H. (eds.) PKAW 2008. LNCS, vol. 5465, pp. 231–241. Springer, Heidelberg (2009)
11. Leggett, J., Williams, G.: Verifying Identity via Keystroke Characteristics. *International Journal of Man-Machine Studies* 28, 67–76 (1988)
12. Joyce, R., Gupta, G.: Identity authentication based on keystroke latencies. *Communications of the ACM* 33(2), 168–176 (1990)
13. de Ru, W.G., Eloff, J.H.P.: Enhanced password authentication through fuzzy logic. *IEEE Expert* 12(6), 38–45 (1997)
14. Haider, S., Abbas, A., Zaidi, A.K.: A multi-technique approach for user identification through keystroke dynamics. In: *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC 2000)*, Nashville, Tenn, USA, vol. 2, pp. 1336–1341 (October 2000)
15. Araújo, L.C.F., Sucupira Jr., L.H.R., Lizárraga, M.G., Ling, L.L., Yabu-Uti, J.B.T.: User authentication through typing biometrics features. *IEEE Transactions on Signal Processing*, part 2 53(2), 851–855 (2005)
16. Fast Artificial Neural Network Library (FANN), <http://leenissen.dk/fann/>

A Fast Incremental Kernel Principal Component Analysis for Online Feature Extraction

Seiichi Ozawa, Yohei Takenchi, and Shigeo Abe

Graduate School of Engineering, Kobe University, Kobe 657-8501, Japan
ozawasei@kobe-u.ac.jp

Abstract. In this paper, we present a modified version of Incremental Kernel Principal Component Analysis (IKPCA) which was originally proposed by Takeuchi et al. as an online nonlinear feature extraction method. The proposed IKPCA learns a high-dimensional feature space incrementally by solving an eigenvalue problem whose matrix size is given by the power of the number of independent data. In the proposed IKPCA, independent data are used for calculating eigenvectors in a feature space, but they are selected in a low-dimensional eigen-feature space. Hence, the size of an eigenvalue problem is usually small, and this allows IKPCA to learn eigen-feature spaces very fast even though the eigenvalue decomposition has to be carried out at every learning stage. The proposed IKPCA consists of two learning phases: initial learning phase and incremental learning phase. In the former, some parameters are optimized and an initial eigen-feature space is computed by applying the conventional KPCA. In the latter, the eigen-feature space is incrementally updated whenever a new data is given. In the experiments, we evaluate the learning time and the approximation accuracies of eigenvectors and eigenvalues. The experimental results demonstrate that the proposed IKPCA learns eigen-feature spaces very fast with good approximation accuracy.

1 Introduction

Eigenspace analysis such as Principal Component Analysis (PCA) has played an important role in classification tasks such as face recognition and object recognition. These methods are used for finding a small number of useful features of target objects, and this feature extraction often enhances the generalization performance of a system as well as the efficiency in memory and computation costs. Recently, Kernel Principal Component Analysis (KPCA) [1] has been extensively studied as an extension of PCA. In KPCA, eigen-axes are obtained in a high-dimensional inner product space called *feature space*. Since KPCA generally gives a set of nonlinear axes in an input space, a complex data distribution can be represented with a small number of such axes; hence, it is expected that this makes the generalization performance of a classifier improved more efficiently. However, KPCA is usually applied to a static data set; therefore, it is not suited for the learning of a dynamic data set, which means that only a small subset of data is given at a time and such subsets are provided sequentially over time. Although the conventional KPCA can be used for incremental learning if all data

are stored in memory, this would be an unrealistic usage for large-scale high-dimensional data such as face images [2]. In this case, an *incremental learning* algorithm for KPCA, which can update an eigenspace model without keeping all the past training data, is solicited under realistic environments.

Many incremental algorithms for eigenspace learning have been proposed so far. Most of them are Incremental PCA (IPCA) [3,4,5,6] or Incremental Linear Discriminant Analysis (ILDA) [7]. To our best knowledge, there have been proposed only a few incremental learning algorithms of KPCA [5,8,9]. This is because the eigenvectors in a feature space cannot be updated in a direct way. To solve this problem, Takenchi et al. proposed an Incremental KPCA (IKPCA) algorithm [9] which was extended from the Incremental PCA (IPCA) algorithm proposed by Hall et al. [3]. In the Takeuchi et al.'s IKPCA, eigenvectors are represented by linearly independent training data which are selected in a low-dimensional eigen-feature space. Therefore, the number of training data to be kept in memory is very small as compared with the conventional IKPCA algorithms [5,8]. This allows the Takeuchi et al.'s IKPCA to learn an eigen-feature space very fast even though the eigenvalue decomposition has to be carried out at every learning stage.

In this paper, we fix the mistakes in the derivation of the update equations on the accumulation ratio in the Takeuchi et al.'s IKPCA, which made the eigenspace learning a little unstable. We further extend the IKPCA algorithm such that parameters are automatically optimized for initial training data based on a cross-validation method. The proposed IKPCA consists of two learning phases: initial learning phase and incremental learning phase. In the former phase, some parameters are optimized and an initial eigen-feature space is computed by applying the conventional KPCA. In the latter phase, the eigen-feature space is incrementally updated whenever a new training data is given.

The rest of this paper is organized as follows. Section 2 gives a brief review on KPCA. In Section 3, we present a modified algorithm for the Takenchi et al.'s IKPCA which had some mistakes in the algorithm derivation. Section 4 shows the experimental results to verify the effectiveness of IKPCA under incremental learning environments. Finally, we give conclusions in Section 5.

2 Kernel Principal Component Analysis

In KPCA, an n -dimensional input \mathbf{x} is mapped to an l -dimensional vector $\phi(\mathbf{x})$ where $\phi(\cdot)$ is the function that maps an input into the l -dimensional feature space. To obtain eigenvectors in the feature space, first we define the following covariance matrix:

$$\mathbf{Q} = \frac{1}{N} \sum_{i=1}^N (\phi(\mathbf{x}_i) - \mathbf{c})(\phi(\mathbf{x}_i) - \mathbf{c})^T \quad (1)$$

where N is the number of input data and $\mathbf{c} = \frac{1}{N} \sum_{i=1}^N \phi(\mathbf{x}_i)$. The eigenvectors are obtained by solving the following eigenvalue problem:

$$\mathbf{QZ} = \mathbf{ZA} \quad (2)$$

where \mathbf{Z} and \mathbf{A} are an eigenvector matrix and an eigenvalue matrix, respectively. Practically, however, solving this problem is hardly carried out in a direct way because the dimensions of a feature space are generally very high and it could be infinite. To avoid the explicit calculation in the feature space, so-called *kernel trick* is applied.

Without loss of generality, we assume that a set of m linearly independent data $\Phi_m = [\phi(\hat{\mathbf{x}}_1), \dots, \phi(\hat{\mathbf{x}}_m)]$ ($m \leq N$) span a feature space where N training data $\{\phi(\mathbf{x}_1), \dots, \phi(\mathbf{x}_N)\}$ are distributed. Then the i th eigenvector \mathbf{z}_i is represented by

$$\mathbf{z}_i = [\phi(\hat{\mathbf{x}}_1), \dots, \phi(\hat{\mathbf{x}}_m)] \begin{bmatrix} \alpha_{1i} \\ \vdots \\ \alpha_{mi} \end{bmatrix} = \Phi_m \alpha_i \tag{3}$$

where $\alpha_i = [\alpha_{1i}, \dots, \alpha_{mi}]^T$ ($i = 1, \dots, m$) is a coefficient vector. Let us define the coefficient matrix $\mathbf{A}_m = [\alpha_1, \dots, \alpha_m]$ and the following kernel matrices:

$$\mathbf{H}_{Nm} = \begin{bmatrix} k(\mathbf{x}_1, \hat{\mathbf{x}}_1) & \cdots & k(\mathbf{x}_1, \hat{\mathbf{x}}_m) \\ \vdots & \ddots & \vdots \\ k(\mathbf{x}_N, \hat{\mathbf{x}}_1) & \cdots & k(\mathbf{x}_N, \hat{\mathbf{x}}_m) \end{bmatrix} \tag{4}$$

$$\mathbf{H}_{mm} = \begin{bmatrix} k(\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_1) & \cdots & k(\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_m) \\ \vdots & \ddots & \vdots \\ k(\hat{\mathbf{x}}_m, \hat{\mathbf{x}}_1) & \cdots & k(\hat{\mathbf{x}}_m, \hat{\mathbf{x}}_m) \end{bmatrix} \tag{5}$$

where $k(\cdot)$ is a kernel function and $k(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)$. Substituting Eq. (1) and Eqs.(3)-(5) into Eq. (2), we can derive the following kernel eigenvalue problem [1]:

$$\frac{1}{N} \mathbf{L}^{-1} \mathbf{H}_{Nm}^T (\mathbf{I}_N - \mathbf{1}_N) \mathbf{H}_{Nm} (\mathbf{L}^{-1})^T (\mathbf{L}^T \mathbf{A}_m) = (\mathbf{L}^T \mathbf{A}_m) \mathbf{A}_m \tag{6}$$

where $\mathbf{L}^T \alpha_i$ (i.e., the i th column vector of $\mathbf{L}^T \mathbf{A}_m$) is the i th eigenvector spanning a feature space and λ_i is the corresponding eigenvalue; \mathbf{I}_N is the $N \times N$ unit matrix and $\mathbf{1}_N$ is the $N \times N$ matrix whose elements are all $1/N$. Here, \mathbf{L} is obtained by the Cholesky factorization for \mathbf{H}_{mm} (i.e., $\mathbf{H}_{mm} = \mathbf{L}\mathbf{L}^T$).

Next assume that we select the first d principal components from the feature space. As a criterion of selecting these components, the following *accumulation ratio* is often adopted:

$$C(d) = \frac{\sum_{i=1}^d \lambda_i}{\sum_{i=1}^m \lambda_i} \tag{7}$$

The accumulation ratio $C(d)$ shows how much information remains in the eigen-feature space after the d components are selected. The dimensionality d is selected such that the accumulation ratio for the d -dimensional eigen-feature space is larger than a certain threshold θ .

3 Incremental Kernel Principal Component Analysis (IKPCA)

The proposed Incremental Kernel PCA (IKPCA) is also derived from the eigenvalue problem in Eq. (2) where a covariance matrix \mathbf{Q} in Eq. (1) is included. However, as in the derivation of KPCA, the matrix decomposition of \mathbf{Q} is not performed in a direct way because the size of a covariance matrix is $l \times l$ where l is very large in general. From Eq. (6), the matrix size for KPCA is actually reduced to $m \times m$ where m is the number of independent data in a feature space.

Although the eigenvalue decomposition of the left-hand side matrix in Eq. (6) is feasible to carry out, the computation costs could increase under incremental learning settings [9]. To make IKPCA more efficient, Takeuchi et al. [9] proposed an improved IKPCA. In this IKPCA, the matrix size is further reduced to $d \times d$ where d is the dimensions of an eigen-feature space that are usually much smaller than l , especially when the RBF kernel is used. In the derivation of the Takeuchi et al.'s IKPCA, we should note that eigenvectors in a feature space are not explicitly calculated; thus, every computation in a feature space should be transformed into a feasible form based on the so-called *kernel trick*.

The learning is divided into two phases: initial learning phase and incremental learning phase. In the former phase, some parameters are optimized and an initial eigen-feature space is computed by performing the conventional KPCA. In the latter phase, the eigen-feature space is incrementally updated whenever a new training data is given. In the following, let us explain the two learning phases in more detail.

3.1 Initial Learning Phase

Assume that N training data $\mathbf{X}_0 = \{\mathbf{x}_i\}_{i=1}^N$ are given with their class information at the initial learning stage. Let us adopt the following RBF kernel function here:

$$k(\mathbf{x}, \mathbf{x}') = \exp(-\gamma \|\mathbf{x} - \mathbf{x}'\|^2) \quad (8)$$

where γ is the kernel parameter. In this case, there are two parameters to be optimized: the threshold θ for the accumulation ratio in Eq. (7) and the kernel parameter γ in Eq. (8).

The purpose of the initial learning phase is not only to compute an initial eigen-feature space but also to find appropriate values of θ and γ . The former computation is basically carried out by applying KPCA to initial training data \mathbf{X}_0 , and the latter operation is conducted with a cross-validation method. The pseudo-code of the initial learning phase is shown in Algorithm 1.

As shown in Algorithm 1, the first procedure is to find optimal values of θ and γ from a candidate set using a cross-validation method. If we adopt the k -fold cross-validation, the following procedures are conducted for every pair of θ and γ . First, \mathbf{X}_0 is divided into k subsets. The $(k - 1)$ subsets are used for training and the remaining one is used for test. The conventional KPCA is applied to the training data to obtain an eigen-feature space model, and the prototypes of the nearest neighbor classifier are obtained by projecting a certain

Algorithm 1. Learn Initial Eigen-feature Space

Input: Training data $\mathbf{X}_N = \{\mathbf{x}_i\}_{i=1}^N$.

Output: Eigen-feature space model $\Omega = \{\mathbf{X}_d, \mathbf{A}_d, \mathbf{A}_m, \Phi_d^T \mathbf{c}, \|\mathbf{c}\|^2, N\}$, threshold θ , kernel parameter γ .

- 1: Perform a cross-validation method using \mathbf{X}_N to find optimal values of θ and γ .
 - 2: Select m data $\mathbf{X}_m = \{\hat{\mathbf{x}}_i\}_{i=1}^m$ from \mathbf{X}_N such that the data in a feature space $\Phi_m = \{\phi(\hat{\mathbf{x}}_i)\}_{i=1}^m$ are linearly independent.
 - 3: Solve the eigenvalue problem in Eq. (6) w.r.t. Φ_m to obtain \mathbf{A}_m and \mathbf{A}_m .
 - 4: Obtain the minimum d such that the accumulation ratio in Eq. (7) holds $C(d) \geq \theta$.
 - 5: Define a coefficient matrix \mathbf{A}_d that consists of the first d column vectors of \mathbf{A}_m .
 - 6: Select d independent data $\Phi_d = \{\phi(\hat{\mathbf{x}}_i)\}_{i=1}^d$ such that \mathbf{D} in Eq. (9) is full rank.
 - 7: Solve the eigenvalue problem in Eq. (6) w.r.t. Φ_d to obtain \mathbf{A}_d and \mathbf{A}_d .
 - 8: Calculate $\|\mathbf{c}\|^2$ and $\Phi_d^T \mathbf{c}$ in Eqs. (10) and (11).
-

number of training data to the eigen-feature space. Then, the test data are projected to the eigen-feature space, and the recognition accuracy is calculated based on the nearest neighbor method. The above procedure is repeated for the k combinations of training and test subsets to estimate the average recognition performance. Finally, the values of θ and γ with the highest average recognition accuracy are selected.

After determining θ and γ , linearly independent data in a feature space are selected from \mathbf{X}_0 . Let the number of such independent data be m and the set of independent data be $\Phi_m = \{\phi(\hat{\mathbf{x}}_i)\}_{i=1}^m$ ¹. Then, the eigenvalue problem in Eq. (6) is solved to obtain the coefficient matrix \mathbf{A}_m and the eigenvalue matrix \mathbf{A}_m . To determine the dimensions of an eigen-feature space, the minimum d is found such that the accumulation ratio $C(d)$ in Eq. (7) is larger than or equal to the threshold θ . After redefining the coefficient matrix \mathbf{A}_d by taking the first d column vectors of \mathbf{A}_m , d linearly independent data $\mathbf{X}_d = \{\hat{\mathbf{x}}_i\}_{i=1}^d$ are selected such that the following kernel matrix \mathbf{D} defined from the projection of $\Phi_m = \{\phi(\hat{\mathbf{x}}_i)\}_{i=1}^m$ to the eigen-feature space is full rank.

$$\mathbf{D} = \mathbf{A}_d^T \left[\Phi_m^T (\phi(\hat{\mathbf{x}}_1) - \mathbf{c}), \dots, \Phi_m^T (\phi(\hat{\mathbf{x}}_m) - \mathbf{c}) \right] \tag{9}$$

where

$$\Phi_m^T \phi(\hat{\mathbf{x}}_i) = [k(\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_i), \dots, k(\hat{\mathbf{x}}_m, \hat{\mathbf{x}}_i)]^T$$

$$\Phi_m^T \mathbf{c} = \frac{1}{N} \left[\sum_{i=1}^N k(\hat{\mathbf{x}}_1, \mathbf{x}_i), \dots, \sum_{i=1}^N k(\hat{\mathbf{x}}_m, \mathbf{x}_i) \right]^T.$$

Then, the eigenvalue problem in Eq. (6) w.r.t. Φ_d is solved to recalculate the coefficient matrix \mathbf{A}_d and the eigenvalue matrix \mathbf{A}_d . Since a data mean \mathbf{c} cannot

¹ Note that the data set $\mathbf{X}_m = \{\hat{\mathbf{x}}_i\}_{i=1}^m$ is actually kept in memory instead of Φ_m .

be held in an explicit form, the following two terms on \mathbf{c} are calculated and kept for the next incremental round:

$$\|\mathbf{c}\|^2 = \frac{1}{N^2} \sum_{i=1}^N \sum_{j=1}^N \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) \quad (10)$$

$$\Phi_d^T \mathbf{c} = \frac{1}{N} \left[\sum_{i=1}^N k(\hat{\mathbf{x}}_1, \mathbf{x}_i), \dots, \sum_{i=1}^N k(\hat{\mathbf{x}}_d, \mathbf{x}_i) \right]^T. \quad (11)$$

Let us denote the calculated eigen-feature space model by the following sextuple:

$$\Omega = \{\mathbf{X}_d, \mathbf{A}_d, \mathbf{A}_d, \Phi_d^T \mathbf{c}, \|\mathbf{c}\|^2, N\}.$$

Note that only d training data \mathbf{X}_d are kept in memory to update an eigen-feature space incrementally.

3.2 Incremental Learning Phase

The pseudo-code of the main procedure in IKPCA is shown in Algorithm 2. After finishing the initial learning phase, the incremental learning is conducted whenever a new training data \mathbf{x} is given.

At first, the numerator and the denominator of $C(d)$ in Eq. (7) are updated as follows²:

$$\sum_{i=1}^d \lambda'_i = \frac{N^2}{(N+1)^3} \left[\frac{(N+1)^2}{N} \sum_{i=1}^d \lambda_i + \sum_{i=1}^d \left\{ \alpha_i^T \left(\Phi_d^T \phi(\mathbf{x}) - \Phi_d^T \mathbf{c} \right) \right\}^2 \right] \quad (12)$$

$$\sum_{i=1}^m \lambda'_i = \frac{N^2}{(N+1)^3} \left[\frac{(N+1)^2}{N} \sum_{i=1}^m \lambda_i + (\phi(\mathbf{x})^T \phi(\mathbf{x}) - 2\phi(\mathbf{x})^T \mathbf{c} + \|\mathbf{c}\|^2) \right]. \quad (13)$$

If $C'(d) \geq \theta$ is satisfied, the given data \mathbf{x} is well represented by the current d -dimensional eigen-feature space. Therefore, the eigen-feature space model Ω is updated without increasing the dimensions to adapt the new data \mathbf{x} . This is done by solving the following eigenvalue problem:

$$\frac{N}{N+1} \left(\mathbf{A}_d + \frac{\mathbf{g}\mathbf{g}^T}{N+1} \right) \mathbf{R} = \mathbf{R}\mathbf{A}'_d \quad (14)$$

where \mathbf{A}'_d and \mathbf{R} respectively correspond to a new eigenvalue matrix and a rotation matrix; \mathbf{g} is given as follows:

$$\mathbf{g} = \begin{bmatrix} \alpha_1^T \left(\mathbf{K}_d(\mathbf{x}) - \Phi_d^T \mathbf{c} \right) \\ \vdots \\ \alpha_d^T \left(\mathbf{K}_d(\mathbf{x}) - \Phi_d^T \mathbf{c} \right) \end{bmatrix}. \quad (15)$$

² In the previous work [9], we had wrong update equations of the accumulation ratio.

Algorithm 2. Incremental Kernel Principal Component Analysis (IKPCA)

Input: Initial training data $\mathbf{X}_N = \{\mathbf{x}_i\}_{i=1}^N$.
Output: Eigen-feature space model $\Omega = \{\mathbf{X}_d, \mathbf{A}_d, \mathbf{A}_d, \Phi_d^T \mathbf{c}, \|\mathbf{c}\|^2, N\}$.
1: // Initial Learning Phase
2: Perform *Learn Initial Eigen-feature Space*.
3: // Incremental Learning Phase.
4: **loop**
5: **Input:** Training data \mathbf{x} .
6: Update $C'(d)$ using Eqs. (7), (12), (13).
7: **if** $C'(d) \geq \theta$ **then**
8: Solve the eigenvalue problem in Eq. (14) to obtain \mathbf{A}'_d and \mathbf{R} .
9: **else**
10: Solve the eigenvalue problem in Eq. (17) to obtain \mathbf{A}'_{d+1} and \mathbf{R} .
11: Add $\phi(\mathbf{x})$ into the independent data set: $\Phi_{d+1} \leftarrow [\Phi_d \ \phi(\mathbf{x})]$.
12: Calculate f^2 using Eq. (18).
13: Update $C'(d+1)$ by adding f^2 to the numerator of $C'(d)$ in Eq. (7).
14: Increment the eigen-feature space dimensions: $d \leftarrow d + 1$.
15: **end if**
16: Update the coefficient matrix \mathbf{A}_d using Eq. (19).
17: Update $\|\mathbf{c}\|^2$ and $\Phi_d^T \mathbf{c}$ using Eqs (20), (21).
18: Increment the number of data: $N \leftarrow N + 1$.
19: **end loop**

On the other hand, if the accumulation ratio $C'(d)$ is smaller than θ , it means the given data \mathbf{x} includes a certain amount of energy in the complimentary eigen-feature space. Therefore, the dimensions of the eigen-feature space should be augmented in the direction of the following residue vector \mathbf{h} :

$$\mathbf{h} \approx [\Phi_d \ \phi(\mathbf{x})] \begin{bmatrix} -\sum_{i=1}^d (\mathbf{K}_d(\mathbf{x})^T \alpha_i) \alpha_i \\ 1 \end{bmatrix}. \tag{16}$$

In order to update the eigen-feature space, the following eigenvalue problem is solved:

$$\frac{N}{N+1} \left(\begin{bmatrix} \mathbf{A}_d & \mathbf{0} \\ \mathbf{0}^T & 0 \end{bmatrix} + \frac{1}{N+1} \begin{bmatrix} gg^T & fg \\ fg^T & f^2 \end{bmatrix} \right) \mathbf{R} = \mathbf{R} \mathbf{A}'_{d+1} \tag{17}$$

where

$$f = \frac{1}{\|\mathbf{h}\|} \left\{ -\sum_{i=1}^d (\mathbf{K}_d(\mathbf{x})^T \alpha_i) \alpha_i^T \left(\mathbf{K}_d(\mathbf{x}) - \Phi_d^T \mathbf{c} \right) + k(\mathbf{x}, \mathbf{x}) - \beta^T (\Phi_d^T \mathbf{c}) \right\}. \tag{18}$$

From Eq. (16), in order to represent \mathbf{h} , the training data $\phi(\mathbf{x})$ should be added to the linearly independent data set. Hence, Φ_d should be update as follows: $\Phi_{d+1} \leftarrow [\Phi_d \ \phi(\mathbf{x})]$. Furthermore, the accumulation ratio $C'(d+1)$ should be recalculated after adding the new eigen-axis. This can be done by adding f^2 to the numerator in Eq. (7).

After calculating the rotation matrix \mathbf{R} , all the eigenvectors are rotated. The rotation of eigenvectors can be equivalently conducted by updating the coefficient matrix as follows:

$$\mathbf{A}'_{d'} = \mathbf{A}_{d'} \mathbf{R} \quad (19)$$

where d' is equal to $d + 1$ if the dimensional augmentation occurs. Finally, the two terms on \mathbf{c} are updated as follows:

$$\|\mathbf{c}'\|^2 = \frac{N^2}{(N+1)^2} \left(\|\mathbf{c}\|^2 + \frac{2}{N} \boldsymbol{\beta}^T (\boldsymbol{\Phi}_d^T \mathbf{c}) + \frac{k(\mathbf{x}, \mathbf{x})}{N^2} \right) \quad (20)$$

$$\boldsymbol{\Phi}_d^T \mathbf{c}' = \frac{1}{N+1} \left\{ N(\boldsymbol{\Phi}_d^T \mathbf{c}) + \mathbf{K}_d(\mathbf{x}) \right\}. \quad (21)$$

4 Performance Evaluation

4.1 Experimental Setup

In this section, we evaluate how well the proposed IKPCA works as an online feature extraction method. For this purpose, we adopt the following two performance scales: (1) learning time and (2) learning accuracy of eigenspaces. The learning time is defined as the time to finish learning a sequence of all training data. The learning accuracy is evaluated using the average direction cosine between two corresponding eigenvectors. Ideally, the eigenvectors of the proposed IKPCA are equivalent with those of KPCA in which all the training data are simultaneously trained in a batch. Therefore, the direction cosine between the two corresponding eigenvectors of IKPCA and KPCA is evaluated to see the identity of eigenspaces.

Six data sets are selected from the UCI Machine Learning Repository [11]. The information on these data sets is shown in Table 1. For the Vowel-context, Adult, and Advertisement data sets, we randomly select up to 1,000 data from the original training data. For the other data sets, training and test data are not separated; thus, we randomly select 1,000 data from the whole data.

At the initial learning stage, 10% of training data are randomly selected as initial training data. The remaining 90% are given to IKPCA one by one.

Table 1. Evaluated data sets

	# Attributes	# Classes	# Training Data
Vowel-context	10	10	528
Adult	14	2	1,000
Segmentation	19	7	1,000
Landsat	36	7	1,000
Ozone	72	2	1,000
Advertisement	1,558	2	1,000

Table 2. Averages and standard deviations of learning time (sec.). The results for KPCA show the time to learn all data in a batch.

	KPCA	CS-IKPCA	IKPCA
Vowel-context	5.7 ± 2.4	80.1 ± 3.2	0.33 ± 0.18
Adult	28.2 ± 11.1	339 ± 25	2.5 ± 10.2
Segmentation	21.8 ± 4.3	622 ± 43	2.1 ± 3.3
Landsat	225 ± 218	259 ± 18	1.6 ± 2.5
Ozone	392 ± 108	697 ± 34	4.9 ± 16.7
Advertisement	265 ± 116	766 ± 11	37.2 ± 51.3

Since the performance of incremental learning generally depends on the sequence of training data, the experiments are carried out for 50 different training sequences to evaluate the average performance. The parameters θ and γ in IKPCA are determined by performing the 5-fold cross-validation, and they are selected from the following candidate sets: $\theta = \{80, 85, 90, 95, 99\}\%$, $\gamma = \{0.05, 0.01, 0.005, 0.001, 0.0005, 0.0001, 0.00005, 0.00001\}$.

4.2 Learning Time

Table 2 shows the results of the learning time (sec.) for KPCA, Chin & Suter's IKPCA (CS-IKPCA) [8], and the proposed IKPCA. For CS-IKPCA, the number of eigenvectors r and the number of preimages p should be determined in advance. Here, r is set to the average dimensions of the eigen-feature spaces obtained by KPCA, and p is set to 10 according to the suggestion in [8]. Moreover, the incremental learning in CS-IKPCA is conducted for every 30 training data because it requires long time to finish learning if training data are given one by one. For KPCA, all the training data are given in a batch to compute eigen-feature vectors.

As shown in Table 2, the learning time of IKPCA is quite shorter than that of KPCA, although the learning of KPCA is conducted in a batch mode (i.e., the number of times to solve an eigenvalue problem is only once). This result also suggests that it is almost unfeasible to use KPCA for an incremental learning purpose. In addition, the proposed IKPCA is also quite faster than CS-IKPCA even though the number of times to solve eigenvalue problems in CS-IKPCA is almost 1/30 as compared with that in IKPCA.

From the above results, we can conclude that the proposed IKPCA can learn very fast under incremental learning settings.

4.3 Learning Accuracy of Eigenspace

The learning accuracy is measured based on the similarities (direction cosines) between two eigenvectors and the following normalized errors:

$$e_i = \frac{|\lambda_i^{\text{bat}} - \lambda_i^{\text{inc}}|}{\sum_{i=1}^d \lambda_i^{\text{bat}}} \quad (22)$$

Table 3. Accuracies of eigen-feature subspaces obtained by IKPCA against KPCA: (a) similarities (direction cosines) of eigenvectors and (b) normalized errors of eigenvalues. The values with bold face fonts correspond to the principal components whose eigenvalues are larger than 5% of the sum of all eigenvalues. The results for the first 10 principal components are shown.

(a)										
	1	2	3	4	5	6	7	8	9	10
Vowel-context	0.9	0.97	0.96	0.96	0.93	0.90	0.94	0.95	0.96	0.92
Adult	1.00	0.93	0.90	0.87	0.87	0.83	0.80	0.78	0.73	0.74
Segmentation	0.99	0.99	0.95	0.94	0.94	0.95	0.91	0.92	0.90	0.89
Landsat	0.99	0.99	0.96	0.87	0.89	0.88	0.92	0.90	0.89	0.83
Ozone	0.99	0.98	0.97	0.94	0.90	0.84	0.85	0.85	0.85	0.83
Advertisement	0.99	0.98	0.96	0.95	0.94	0.94	0.92	0.88	0.87	0.89

(b)										
	1	2	3	4	5	6	7	8	9	10
Vowel-context	0.018	0.013	0.016	0.009	0.006	0.005	0.003	0.002	0.001	0.002
Adult	0.024	0.009	0.008	0.007	0.006	0.007	0.006	0.007	0.007	0.008
Segmentation	0.008	0.003	0.006	0.003	0.003	0.002	0.002	0.002	0.002	0.001
Landsat	0.013	0.018	0.006	0.004	0.002	0.002	0.002	0.002	0.001	0.001
Ozone	0.014	0.012	0.008	0.006	0.006	0.006	0.005	0.004	0.004	0.004
Advertisement	0.012	0.009	0.007	0.007	0.006	0.004	0.004	0.004	0.004	0.004

where λ_i^{bat} and λ_i^{inc} are the i th eigenvalues calculated by KPCA and IKPCA, respectively.

Tables 3 (a) and (b) show the similarities of eigenvectors and the normalized errors of eigenvalues for the first 10 principal components, respectively. Here, the eigenvectors whose eigenvalue is larger than 5% of the sum of all eigenvalues are called *major components* and the results for the major components are shown in a bold font. From Tables 3 (a) and (b), the major components are well approximated with high similarities (over 0.9) except for the Adult data, and the normalized errors are less than 2.5%. From these results, we conclude that the proposed IKPCA can approximate eigenspaces with good accuracy.

5 Conclusions

In this paper, we fix some mistakes in the derivation of the Takeuchi et al.'s IKPCA [9] which made the eigenspace learning a little unstable. In addition, we extend the IKPCA algorithm such that parameters are automatically optimized for initial training data using a cross-validation method. The proposed IKPCA consists of the two learning phases: initial learning phase and incremental learning phase. In the former phase, the threshold of the accumulation ratio and the kernel parameter are optimized, and then an initial eigen-feature space is computed by applying the conventional KPCA. In the latter phase, the eigen-feature space is incrementally updated whenever a new training data is given.

The proposed IKPCA learns a high-dimensional feature space incrementally by solving an eigenvalue problem whose matrix size is given by the power of the number of independent data. Since independent data are selected in a low-dimensional eigen-feature space spanned by eigenvectors, the matrix size in the eigenvalue problem is generally small, and this allows IKPCA to learn an eigen-feature space very fast even though the eigenvalue decomposition has to be carried out at every learning stage.

To verify the effectiveness of the proposed IKPCA, the learning time and the accuracies of eigenvectors and eigenvalues are evaluated for the six benchmark data sets in the UCI machine learning repository. The experimental results show that the proposed IKPCA can learn an eigen-feature space very fast compared with the Chin & Suter's IKPCA, and accurate eigenvectors and eigenvalues are obtained especially for major components.

References

1. Abe, S.: *Support Vector Machines for Pattern Classification*. Springer, London (2005)
2. Ozawa, S., Toh, S.L., Abe, S., Pang, S., Kasabov, N.: Incremental Learning of Feature Space and Classifier for Face Recognition. *Neural Networks* 18, 575–584 (2005)
3. Hall, P., Marshall, D., Martin, R.: Incremental Eigenanalysis for Classification. In: *Proc. of British Machine Vision Conference*, pp. 286–295 (1998)
4. Weng, J., Zhang, Y., Hwang, W.S.: Candid Covariance-free Incremental Principal Component Analysis. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 25, 1034–1040 (2003)
5. Zhao, H., Chi, P., Kwok, J.T.: A Novel Incremental Principal Component Analysis and Its Application for Face Recognition. *IEEE Trans. on Systems, Man and Cybern., Part B* 36, 873–886 (2006)
6. Ozawa, S., Pang, S., Kasabov, N.: Incremental Learning of Chunk Data for Online Pattern Classification Systems. *IEEE Trans. on Neural Networks* 19, 1061–1074 (2008)
7. Pang, S., Ozawa, S., Kasabov, N.: Incremental Linear Discriminant Analysis for Classification of Data Streams. *IEEE Trans. on Systems, Man and Cybern., Part B* 35, 905–914 (2005)
8. Chin, T.-J., Suter, D.: Incremental Kernel Principal Component Analysis. *IEEE Trans. on Image Processing* 16, 1662–1674 (2007)
9. Takenchi, Y., Ozawa, S., Abe, S.: An Efficient Incremental Kernel Principal Component Analysis for Online Feature Selection. In: *Proc. Int. Joint Conf. on Neural Networks*, pp. 1603–1608 (2007)
10. Kim, B.-J.: Active Visual Learning and Recognition Using Incremental Kernel PCA. In: Zhang, S., Jarvis, R.A. (eds.) *AI 2005. LNCS (LNAI)*, vol. 3809, pp. 585–592. Springer, Heidelberg (2005)
11. <http://www.ics.uci.edu/~mlearn/MLRepository.html>

Colour Object Classification Using the Fusion of Visible and Near-Infrared Spectra

Heesang Shin¹, Napoleon H. Reyes¹, Andre L. Barczak¹, and Chee Seng Chan²

¹ Institute of Information and Mathematical Sciences, Massey University, Auckland
New Zealand

² Mimos Berhad, Kuala Lumpur, Malaysia

H.Shin@massey.ac.nz, N.H.Reyes@massey.ac.nz, A.L.Barczak@massey.ac.nz,
CS.Chan@mimos.my

<http://www.massey.ac.nz/~hshin>

Abstract. Under extreme light conditions, a conventional colour CCD camera would fail to render the colours of an object properly as the visible spectrum is either faintly observable in the scene or the presence of glare corrupts the colours sensed. On the other hand, for darkly-illuminated areas, a near-infrared (NIR) camera would sense stronger more discriminable signals, but could only render the scene monochromatically. The underlying challenge in this research is how to adaptively integrate a monochromatic NIR image with a faintly rendered colour image of the same darkly or very brightly lit scene to give rise to improved colour classification results that discriminate colours more effectively. This research proposes a Fuzzy-Genetic colour processing algorithm that adaptively marries together the visible and near-infrared spectra signals for the purpose of colour object recognition. The experiments were done on a scene with spatially varying illumination intensities, using Fujifilm's UV/IR Super CCD camera with a sensitivity range between 380nm to 1000nm in conjunction with NIR filters. Results prove that the proposed multi-spectrum technique yields better colour classification results than utilizing the pure visible spectrum alone.

1 Introduction

There is a breaking point for colour classification techniques operating within the limits of the visible spectrum. For very dark exploratory regions, only the longer wavelengths of light are mostly present in the scene, while the others significantly fade. On the other hand, the presence of glare causes the pixel colours to approach pure white. In the electromagnetic spectrum, there is a region that corresponds to the non-visible, infrared spectrum (0.7-2.4 micro meter) [1] that is not yet fully explored for colour classification. It can be deduced that cultivating these infrared signals and integrating them with the colour sensed values in the visible spectrum will expand the colour discrimination capabilities of computer vision systems. However, the integration of the signals is by far non-trivial and also requires that the fused colours be discriminable despite the presence of gradients in the illumination intensities. In addition, similarly coloured objects (e.g.

orange, red, pink, violet) should be distinguishable from each other, regardless of their position in the exploratory field (i.e. dim, dark or bright illumination setting). Therefore, the ultimate goal of the fusion of visible and infrared signals is to allow for adaptive colour correction to improve colour classification under spatially varying illumination intensities. Due to the limitations of the camera used, the scope of this work only explores the integration of the visible (480 - 700 nm) and near infrared spectra (700-900 nm).

Innate in the human visual system is our ability to compensate for the effects of illumination changes, allowing us to perceive the colours of objects more stably. This capability is known as colour constancy [2], and is a desirable feature for any colour object recognition system. There were many attempts to mimic this capability computationally, but most colour constancy algorithms operate with great efficacy only on scenes with uniform illumination condition [2]. In general, colour constancy algorithms aim to keep constant the computed colour of an image pixel irrespective of the illumination present in the scene [2]. On the contrary, the proposed algorithms in this paper aim to keep constant the position (i.e. Cartesian coordinates) of the computed colour of an image pixel in the colour space, within the confines of a pie-slice decision region assigned to it for its classification. Within a scene, the proposed algorithm performs colour correction only on the candidate pixels depicting the target colours to be tracked down; the rest of the colours in the image remain unscathed. We call this technique selective colour constancy. The colour corrections are employed not to improve the appearance of the colours per se, but with the aim of classifying the target colours more accurately. Multispectral selective colour constancy in this research is achieved by means of a Fuzzy-Genetic colour contrast fusion that adaptively enhances or degrades the colour tristimulus, thereby influencing the formation of colours depicting the target object within a pie-slice decision region in the rg-chromaticity colour space.

2 Related Works

The use of multispectral imaging has come of age to be a viable alternative to conventional broadband monochromatic or colour imaging cameras for a multitude of imaging applications: face recognition with different poses and expressions [3], geographical studies [4,5], food processing industry [6,7,8,9,10] and medical imaging [11,12].

Multispectral imaging captures a wide range of light reflectance and thermal radiation information, spanning both the visible and near infrared spectra (non-visible). The general technique employed in multi-spectral imaging requires a set of images, each acquired at a narrow band of wavelengths. Using a UV/IR camera with a bandpass filter (or interference filter) in front of it, images could be obtained at discrete spectral regions [8]. Vilaseca et al. [13] introduced multiple pseudo colour schemes in NIR to colourise these discrete spectral regions. On the other hand, some studies employed the whole spectral range as input. Menesatti et al. [14] used monochromatic spectrophotometer for VIS to NIR

spectrum range to analyse plant nutritional status. Mertens et al. [15] also used a combination of VIS to NIR spectrum range to analyse egg shell colours for quality measure using the L*a*b colour space. In contrast, Pap and Žiljak studied the separation of the near infrared wavelength area in case of a double image reproduction [16].

All of the aforementioned undertakings capitalize on the visible and near infrared spectra integration to extract useful patterns or signatures of objects; they do not however, revive the colours of the objects. There is no study that we know of that tried to combine both complementary spectral ranges into one colour scheme for improved colour classification yet. What makes this research unique is that we propose an adaptive Fuzzy Genetic-based visible and near infrared spectra integration technique with adaptive fuzzy colour enhancement and degradation operators that revive colours in very low light conditions. The Genetic Algorithm component of the system fully-automatically fine-tunes all parameters required by the colour classifiers. Once calibration is completed, colour classification is performed in real-time using a novel variable-depth colour lookup table [17,18].

3 Illustration of the Problem Domain

There is a problem in colour object classification when the colours of an object become indistinguishable due to very strong or very weak illumination conditions, and also due to the limits of the sensitivity range of the colour CCD camera. In this case, it is extremely difficult, if not impossible, to estimate the real colours of the object merely from the colour information captured from the visible spectrum. However, for multispectrum cameras, it is still possible to extract further information from the same pixel location in the near-infrared range. Figure 1 shows an example of a scene with a very low light condition and with illumination gradients. On the other hand, Fig. 2 shows a near-infrared image of the same scene.

4 Experiment Set-Up

A Fujifilm IS Pro UV/IR camera is used for capturing all the multi-spectral images. The camera's sensitivity ranges from 380nm to 1000nm, covering the ultraviolet and near-infrared spectra. Eight different filters were used to control the transmission of light: Peca 902(#70), 904(#87), 906(#87a), 908(#87b), 910(#87c), 912(#88a), 914(#89b) and 918(visible spectrum). The numbers in the parenthesis corresponds to the Wratten optical filter label. The exposure time used are 200, 167, 125, 100, 77, 67, 50 and 40 milliseconds - this simulates the different ambient lighting conditions. The light source is a standard halogen spotlight with a 50-Watt capacity. Seven target colour patches, each with 4 representatives were classified. These colour patches were strategically placed in varying illumination intensities to test for relatively bright, dim and dark illumination conditions.

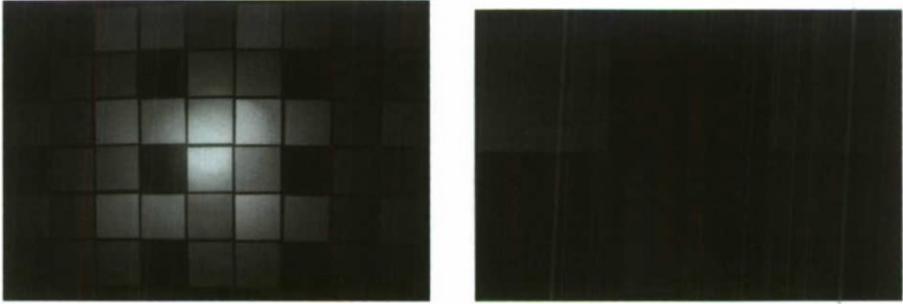


Fig. 1. An example of a scene with spatially varying illumination intensities reflecting the visible spectrum. The image on the right is an enlargement of the upper right corner section of the same scene. Fujifilm IS Pro, F/3.5, 1/20 sec, ISO 100, Peca 918 Filter

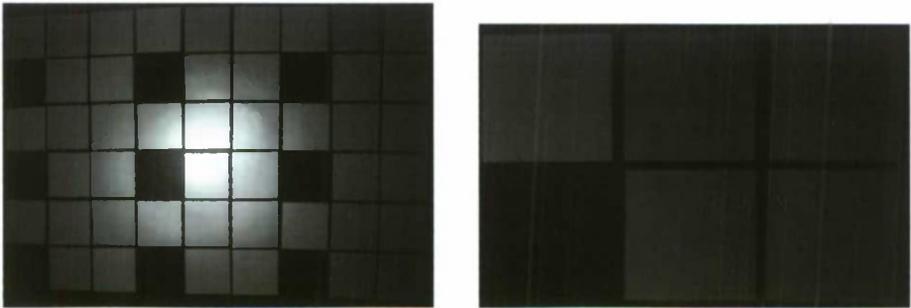


Fig. 2. The corresponding image of the scene shown in Fig. 1, reflecting only the near infrared spectrum. The image on the right is an enlargement of the upper right corner section of the same scene. Fujifilm IS Pro, F/3.5, 1/20 sec, ISO 100, Peca 904 Filter.

5 The Algorithms

5.1 Colour Space and the Pie-Slice Decision Region

The fused visible and NIR signals are scaled to be representable in a modified rg-chromaticity colour space, where the colour descriptors used [19] are suitable for pie-slice colour classification [20]: rg-Hue corresponds to the angle, while rg-Saturation corresponds to the radius of a colour pixel.

$$\text{rg-chromaticities: } r = \frac{R}{R+G+B}, g = \frac{G}{R+G+B}$$

$$\text{rg-Saturation} = \sqrt{(r - 0.333)^2 + (g - 0.333)^2}, \text{ rg-Hue} = \tan^{-1}\left(\frac{g - 0.333}{r - 0.333}\right)$$

5.2 Fuzzy Colour Contrast Fusion (FCCF)

The resulting geometric shape of the distribution of the fused colour pixel values depicting the target colour objects is not readily amenable for colour classification using a pie-slice decision region in the rg-chromaticity space. The drifting of the colour pixel values in the colour space is highly non-linear due to the effects of spatially varying illumination intensities and this is compensated for by a fuzzy colour processing algorithm called FCCF [19], in combination with a Heuristically Assisted Genetic Algorithm (HAGA) [21] for automatically extracting the parameters of the colour classifiers.

The inputs to FCCF are the combined visible and NIR colour tristimulus in RGB form, as well as the calculated rg-Hue and rg-Saturation values. HAGA instructs FCCF how to operate on the raw input colours by feeding it with the evolved colour classifier parameters. The parameters mainly consist of the set of optimal colour contrast rules for both the visible and near infrared channels, the colour contrast enhance (1) and degradation operations (2), and the colour contrast constraint angles for the fused visible and NIR channels. Consequently, FCCF returns the refined RGB values amenable for final colour classification.

Contrast Enhance Operator:

$$\alpha = \begin{cases} 2\mu_{\alpha}^2(y) & 0 \leq \mu_{\alpha}(y) < 0.5 \\ 1 - 2[1 - \mu_{\alpha}(y)]^2 & 0.5 \leq \mu_{\alpha}(y) \leq 1 \end{cases} \quad (1)$$

Contrast Degrade Operator:

$$\alpha = \begin{cases} 0.5 + 2[\mu_{\alpha}(y) - 0.5]^2 & 0 \leq \mu_{\alpha}(y) < 0.5 \\ 0.5 - 2(1 - [\mu_{\alpha}(y) + 0.5]^2) & 0.5 \leq \mu_{\alpha}(y) \leq 1 \end{cases} \quad (2)$$

5.3 VIS-NIR Fusion Operators

The fuzzy colour contrasted near-infrared signal is fused adaptively with the visible colour tristimulus according to the fusion operation range. The candidate fusion operators are listed by Equation 3 where α is one of the colour components of the visible spectrum (e.g. R, G or B) and β is the fuzzy colour contrasted value of the NIR signal. For each colour channel (i.e. R,G,B), both the fusion operation, fusion operation range and the colour contrast operator for the NIR signal are selected automatically by the HAGA algorithm.

$$\begin{aligned} & \text{(a) } \alpha = \alpha * (1 + \beta), \quad \text{(b) } \alpha = \alpha * \beta, \text{(c) } \alpha = \alpha - \beta \\ & \text{(d) } \alpha = \frac{(\alpha + \beta)}{2}, \quad \text{(e) } \alpha = \alpha + \beta, \text{(f) } \alpha = \beta \\ & \alpha = 1, \alpha > 1 \\ & \alpha = |\alpha|, \alpha < 0 \end{aligned} \quad (3)$$

Parameter	Range	Length	Incremental Steps
Min Angle	0° ~ 360°	10 bits	0.351
Max Angle	0° ~ 360°	10 bits	0.351
Min Radius	0 ~ 1	10 bits	0.001
Max Radius	0 ~ 1	10 bits	0.001
Min Contrast Angle	0° ~ 360°	10 bits	0.351
Max Contrast Angle	0° ~ 360°	10 bits	0.351

Parameter	Range	Length	Incremental Steps
Red Contrast Rule	-3.00 ~ 3.99	6 bits	0.109
Green Contrast Rule	-3.00 ~ 3.99	6 bits	0.109
Blue Contrast Rule	-3.00 ~ 3.99	6 bits	0.109
Red Colour Depth	5 ~ 8.99	4 bits	0.249
Green Colour Depth	5 ~ 8.99	4 bits	0.249
Blue Colour Depth	5 ~ 8.99	4 bits	0.249

Parameter	Range	Length	Incremental Steps
Red Fusion Operation	0 ~ 8.99	5 bits	0.281
Green Fusion Operation	0 ~ 8.99	5 bits	0.281
Blue Fusion Operation	0 ~ 8.99	5 bits	0.281
Red Fusion Operation Range	-1 ~ 1	7 bits	0.016
Green Fusion Operation Range	-1 ~ 1	7 bits	0.016
Blue Fusion Operation Range	-1 ~ 1	7 bits	0.016
Red Contrast Rule	-3.00 ~ 3.99	6 bits	0.109
Green Contrast Rule	-3.00 ~ 3.99	6 bits	0.109
Blue Contrast Rule	-3.00 ~ 3.99	6 bits	0.109

Fig. 4. Chromosome Design

6.2 Fitness Function

The evolved colour classifiers represented by the chromosomes are automatically graded using a fitness function described in [21]. The fitness function (Eqn. (4)) adaptively forgives false positive classifications to encourage finding classifiers that return high true positives. On the other hand, it tries to avoid getting trapped in local maxima by reducing rewards in cases where true positives and false positives are both very low.

$$\begin{aligned}
 x &= \frac{\text{true positive pixels in target area}}{\text{total pixels in target area}} \\
 y &= \frac{\text{false positive pixels in outside target area}}{\text{total pixels outside target area}} \\
 \text{fitness} &= \frac{1}{2} \left[\frac{1}{e^{-10(y-0.5)}} + \left(\frac{1 - \frac{1}{1+e^{-75(x-0.05)}}}{1 + e^{-10(y-0.4)}} \right) \right]
 \end{aligned} \tag{4}$$

7 Experiment Results and Analysis

Fig. 5 illustrates the colour classification results using the pure visible spectrum and fused visible and NIR spectra. As can be seen from the pure visible spectrum Image (a), pink and violet are hardly distinguishable from each other. On the other hand, by utilising the NIR Image (b) for additional colour information and applying fuzzy colour contrast fusion and colour classifier optimisation by HAGA, the resulting Image (c) dramatically changed in colour as compared to the original one.

What's interesting to see in the results is the revival of the colours of the two Pink colour patches in the centre of the image. This is reflected by the results found in Image (e) - fused visible and NIR classification results, as well as Image

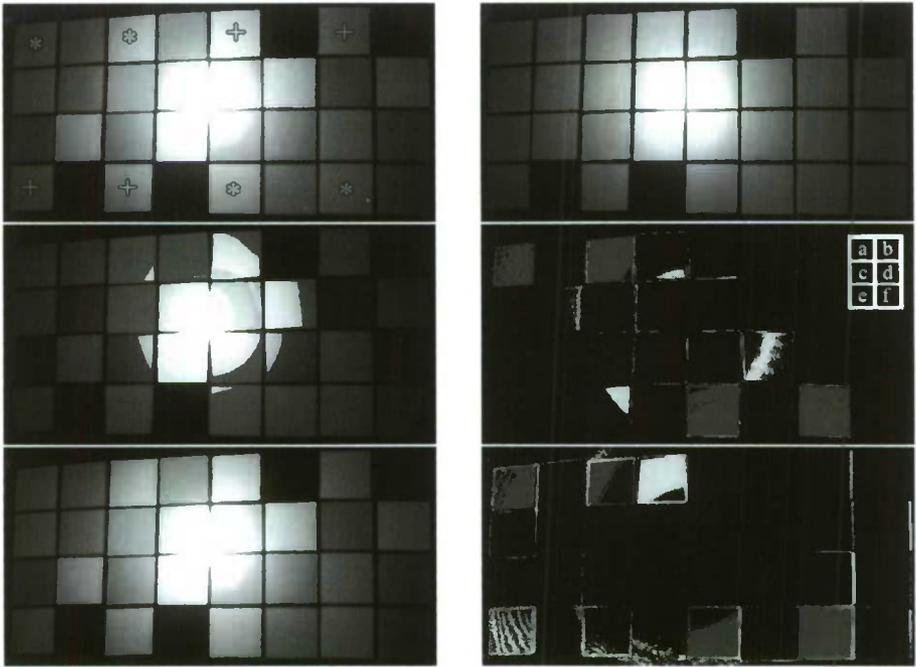


Fig. 5. Sample colour classification results for the Pink target colour patches. The * labels identify the Pink targets and the + labels identify the Violet targets in image (a). (a) original image under 200ms exposure time, using Peca 918 filter. (b) near-infrared image under 200ms exposure time using Peca 908 filter. (c) fused and fuzzy colour contrasted image using images (a) and (b) as inputs. (d) colour classification results using the image in (c), red pixels depict true positive results while yellow depict false positives. (e) fuzzy colour contrasted image of the scene in (a). (f) colour classification results using the image in (e), red pixels depict true positive results while yellow depict false positives.

Table 1. Colour Classification Result: Visible Versus Fusion of Visible and NIR Spectra (Green)

Shutter Speed (ms)	Fusion of Visible and Near Infrared Spectra									Percentage of Improvement	
	VIS	902	904	906	908	910	912	914	FUSION BEST RESULT	BEST FILTER	Over the Visible
200	0.958	0.919	0.803	0.939	0.937	0.956	0.919	0.949	0.956	910	-0.261%
167	0.963	0.961	0.908	0.915	0.914	0.959	0.962	0.925	0.962	912	-0.127%
125	0.965	0.863	0.964	0.756	0.902	0.914	0.735	0.875	0.964	904	-0.076%
100	0.966	0.916	0.921	0.955	0.889	0.958	0.949	0.964	0.964	914	-0.246%
77	0.968	0.898	0.913	0.942	0.925	0.952	0.916	0.932	0.952	910	-1.618%
67	0.967	0.914	0.932	0.920	0.738	0.956	0.938	0.948	0.956	910	-1.155%
50	0.967	0.944	0.933	0.936	0.935	0.919	0.937	0.941	0.944	902	-2.479%
40	0.961	0.734	0.885	0.871	0.953	0.922	0.895	0.872	0.953	908	-0.774%
Average	0.964	0.894	0.907	0.904	0.899	0.942	0.907	0.926	0.956	910	-0.837%

Table 2. Colour Classification Result: Visible Versus Fusion of Visible and NIR Spectra (Light Blue)

Shutter Speed (ms)	Fusion of Visible and Near Infrared Spectra											Percentage of Improvement Over the Visible
	VIS	Filter Type							FUSION BEST	BEST	FILTER	
		902	904	906	908	910	912	914	RESULT			
200	0.971	0.969	0.967	0.970	0.972	0.967	0.968	0.968	0.972	908	0.150%	
167	0.975	0.970	0.973	0.969	0.977	0.975	0.974	0.972	0.977	908	0.249%	
125	0.977	0.973	0.976	0.977	0.975	0.977	0.977	0.975	0.977	912	0.014%	
100	0.978	0.977	0.977	0.977	0.978	0.979	0.974	0.979	0.979	912	0.085%	
77	0.978	0.973	0.978	0.976	0.976	0.977	0.976	0.973	0.978	904	0.037%	
67	0.980	0.979	0.980	0.981	0.978	0.979	0.977	0.978	0.981	906	0.114%	
50	0.980	0.979	0.980	0.979	0.980	0.979	0.979	0.978	0.980	908	-0.003%	
40	0.971	0.967	0.971	0.966	0.970	0.970	0.971	0.969	0.971	904	-0.009%	
Average	0.976	0.973	0.975	0.974	0.976	0.975	0.975	0.974	0.977	908	0.080%	

Table 3. Colour Classification Result: Visible Versus Fusion of Visible and NIR Spectra (Orange)

Shutter Speed (ms)	Fusion of Visible and Near Infrared Spectra											Percentage of Improvement Over the Visible
	VIS	Filter Type							FUSION BEST	BEST	FILTER	
		902	904	906	908	910	912	914	RESULT			
200	0.708	0.497	0.497	0.497	0.850	0.497	0.558	0.497	0.850	908	16.692%	
167	0.941	0.497	0.954	0.497	0.497	0.497	0.734	0.497	0.954	901	1.372%	
125	0.946	0.682	0.497	0.497	0.716	0.497	0.497	0.608	0.716	908	-32.089%	
100	0.939	0.881	0.497	0.561	0.497	0.497	0.497	0.836	0.881	902	-6.670%	
77	0.917	0.497	0.794	0.642	0.823	0.497	0.497	0.497	0.823	908	-11.408%	
67	0.797	0.497	0.497	0.497	0.497	0.497	0.497	0.497	0.497	902	-60.583%	
50	0.618	0.497	0.497	0.497	0.497	0.497	0.497	0.497	0.497	902	-21.513%	
40	0.496	0.496	0.496	0.496	0.496	0.496	0.496	0.496	0.496	914	0.011%	
Average	0.795	0.568	0.591	0.523	0.609	0.497	0.534	0.553	0.714	908	-11.378%	

Table 4. Colour Classification Result: Visible Versus Fusion of Visible and NIR Spectra (Pink)

Shutter Speed (ms)	Fusion of Visible and Near Infrared Spectra											Percentage of Improvement Over the Visible
	VIS	Filter Type							FUSION BEST	BEST	FILTER	
		902	904	906	908	910	912	914	RESULT			
200	0.704	0.673	0.861	0.641	0.931	0.740	0.781	0.933	0.933	914	21.589%	
167	0.869	0.947	0.850	0.937	0.943	0.763	0.851	0.949	0.949	914	8.397%	
125	0.948	0.938	0.903	0.909	0.939	0.894	0.906	0.877	0.939	908	-1.033%	
100	0.927	0.924	0.920	0.959	0.939	0.896	0.933	0.927	0.959	906	3.339%	
77	0.907	0.928	0.937	0.917	0.935	0.952	0.928	0.923	0.952	910	4.795%	
67	0.948	0.893	0.895	0.946	0.932	0.957	0.922	0.920	0.957	910	0.850%	
50	0.930	0.911	0.895	0.943	0.911	0.935	0.898	0.945	0.945	914	1.571%	
40	0.840	0.759	0.847	0.878	0.796	0.839	0.800	0.852	0.878	906	4.280%	
Average	0.884	0.872	0.888	0.891	0.916	0.872	0.877	0.916	0.916	914	3.411%	

(f) - visible spectrum classification results. It is evident that the true positives increased in Image (e) after the fusion of visible and NIR signals with FCCF and HAGA operations.

The experiments involved training the colour classifier using FCCF-HAGA-VCD which takes inputs from the visible and NIR images at 8 different shutter speeds. The visible and NIR images show that the colour objects are under spatially varying illumination conditions. 7 different filters were used for taking

Table 5. Colour Classification Result: Visible Versus Fusion of Visible and NIR Spectra (Violet)

Shutter Speed (ms)	Fusion of Visible and Near Infrared Spectra										Percentage of Improvement Over the Visible
	VIS	902	904	906	908	910	912	914	FUSION BEST RESULT	BEST FILTER	
200	0.886	0.935	0.868	0.938	0.895	0.964	0.886	0.918	0.964	910	8.153%
167	0.883	0.910	0.947	0.950	0.882	0.877	0.896	0.956	0.956	914	7.662%
125	0.927	0.939	0.853	0.936	0.839	0.922	0.921	0.954	0.954	914	2.886%
100	0.935	0.950	0.913	0.937	0.941	0.950	0.937	0.929	0.950	902	1.562%
77	0.973	0.932	0.965	0.960	0.960	0.955	0.917	0.942	0.965	904	-0.815%
67	0.901	0.943	0.945	0.919	0.938	0.916	0.900	0.937	0.945	904	4.598%
50	0.916	0.928	0.933	0.929	0.930	0.931	0.935	0.919	0.935	912	2.089%
40	0.889	0.851	0.871	0.900	0.880	0.854	0.918	0.892	0.918	912	3.190%
Average	0.914	0.923	0.912	0.934	0.908	0.921	0.914	0.931	0.948	906	3.673%

Table 6. Colour Classification Result: Visible Versus Fusion of Visible and NIR Spectra (Red)

Shutter Speed (ms)	Fusion of Visible and Near Infrared Spectra										Percentage of Improvement Over the Visible
	VIS	902	904	906	908	910	912	914	FUSION BEST RESULT	BEST FILTER	
200	0.939	0.932	0.933	0.933	0.936	0.942	0.933	0.933	0.942	910	0.277%
167	0.956	0.934	0.926	0.928	0.934	0.933	0.933	0.939	0.939	914	-1.832%
125	0.968	0.953	0.924	0.959	0.959	0.960	0.946	0.922	0.960	910	-0.802%
100	0.963	0.960	0.954	0.904	0.938	0.962	0.911	0.944	0.962	910	-0.113%
77	0.966	0.966	0.948	0.870	0.948	0.859	0.894	0.890	0.966	902	0.051%
67	0.949	0.892	0.751	0.795	0.747	0.949	0.948	0.917	0.949	910	-0.007%
50	0.921	0.914	0.839	0.911	0.881	0.848	0.746	0.707	0.914	902	-0.780%
40	0.822	0.666	0.496	0.614	0.680	0.816	0.682	0.816	0.816	910	-0.696%
Average	0.936	0.902	0.846	0.864	0.878	0.909	0.874	0.883	0.931	910	-0.480%

Table 7. Colour Classification Result: Visible Versus Fusion of Visible and NIR Spectra (Yellow)

Shutter Speed (ms)	Fusion of Visible and Near Infrared Spectra										Percentage of Improvement Over the Visible
	VIS	902	904	906	908	910	912	914	FUSION BEST RESULT	BEST FILTER	
200	0.943	0.709	0.719	0.811	0.497	0.594	0.497	0.497	0.811	906	-16.276%
167	0.954	0.553	0.497	0.497	0.700	0.716	0.497	0.497	0.716	910	-33.236%
125	0.961	0.497	0.497	0.531	0.899	0.497	0.637	0.675	0.899	908	-6.906%
100	0.965	0.645	0.497	0.703	0.497	0.917	0.497	0.497	0.917	910	-5.171%
77	0.967	0.922	0.958	0.836	0.791	0.766	0.907	0.511	0.958	904	-0.926%
67	0.969	0.958	0.679	0.795	0.926	0.497	0.649	0.752	0.958	902	-1.138%
50	0.968	0.763	0.811	0.497	0.497	0.745	0.497	0.497	0.811	904	-19.341%
40	0.949	0.838	0.496	0.803	0.926	0.496	0.848	0.844	0.926	908	-2.462%
Average	0.960	0.736	0.644	0.684	0.717	0.653	0.628	0.596	0.875	902	-9.705%

the NIR images with varying transmission characteristics. For each of the 7 target colours, 448 colour classifiers were generated and compared.

The details of the classification results can be found in Table 1, 2, 3, 4, 5, 6 and 7. All the results, including the pure visible and the fusion of visible and near-infrared signals were calculated using Equ. (4). The percentage of improvement was calculated based on the best result from the fusion of signals (Fusion Best Result Column) and the best result from the pure visible spectrum (VIS column).

It can be seen from the tabulated results that the classification of Pink, Violet and Light Blue target colours improved over the pure visible approach. For the rest of the other colours no improvement was observed. We hypothesise that the lack of improvement is due to the insufficient amount of generation and population size used by the Genetic Algorithm. The chromosome size for the fused visible and near-infrared is double the chromosome size used for the pure visible approach. It can only be deduced that the increase in chromosome size should be accompanied by a significant increase in generation and population size. We intend to test this hypothesis further in our future work.

8 Conclusion

This research sets foot on the fusion of visible and near-infrared spectra for the purpose of colour classifying objects at spatially varying illumination intensities. Empirical results show that the proposed integration process and the accompanying Fuzzy-Genetic colour processing algorithms can revive colours that are hardly distinguishable from the pure visible spectrum image alone.

Acknowledgement. We would like to thank our colleagues Prof. Tony Norris and Prof. Kenneth Hawick for their unfaltering support. Our thanks also go to Gary Garnett of Peca Products Inc. for his valuable information on the different filters, and Marcus Lacey of Fujifilm NZ Ltd. for helping us acquire the UV/NIR camera. Likewise, we are grateful to Matthew Wall of Massachusetts Institute of Technology for sharing his GA Library, and to the anonymous reviewers of our paper for their insightful comments and suggestions.

References

1. Kong, S., Heo, J., Abidi, B., Paik, J., Abidi, M.: Recent advances in visual and infrared face recognition - a review. *The Journal of Computer Vision and Image Understanding* 97(1), 103–135 (2005)
2. Ebner, M.: *Color Constancy*. Wiley, Chichester (2007)
3. Pan, Z., Healey, G., Prasad, M., Tromberg, B.: Face recognition in hyperspectral images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 25, 1552–1560 (2003)
4. Rauss, P.J., Daida, J.M., Chandhary, S.: Classification of spectral imagery using genetic programming. In: *Proc. GECCO*, pp. 726–733 (2000)
5. Montoliu, R., Pla, F., Klaren, A.C.: Illumination intensity, object geometry and highlights invariance in multispectral imaging (2005)
6. Ghosh, P., Jayas, D.: Use of spectroscopic data for automation in food processing industry. *Sensing and Instrumentation for Food Quality and Safety* 3(1), 3–11 (2009)
7. Chao, K., Park, B., Chen, Y., Hruschka, W., Wheaton, F.: Design of a dual-camera system for poultry carcasses inspection. *Appl. Eng. Agric.* 16(5), 581–587 (2000)
8. Chen, Y.R., Chao, K., Kim, M.S.: Machine vision technology for agricultural applications. *Computers and Electronics in Agriculture* 36(2-3), 173–191 (2002)

9. Kleynen, O., Leemans, V., Destain, M.F.: Development of a multi-spectral vision system for the detection of defects on apples. *Journal of Food Engineering* 69(1), 41–49 (2005)
10. ElMasry, G., Wang, N., Vigneault, C., Qiao, J., ElSayed, A.: Early detection of apple bruises on different background colors using hyperspectral imaging. *LWT - Food Science and Technology* 41(2), 337–345 (2008)
11. Kobayashi, H., Ogawa, M., Kosaka, N., Choyke, P., Urano, Y.: Multicolor imaging of lymphatic function with two nanomaterials: quantum dot-labeled cancer cells and dendrimer-based optical agents. *Nanomedicine* 4, 411–419 (2009)
12. Kosaka, N., Ogawa, M., Longmire, M.R., Choyke, P.L., Kobayashi, H.: Multi-targeted multi-color in vivo optical imaging in a model of disseminated peritoneal ovarian cancer. *Journal of Biomedical Optics* 14 (2009)
13. Vilaseca, M., Pujol, J., Arjona, M., Martinez-Verd, F.M.: Color visualization system for near-infrared multispectral images. *Journal of Imaging Science and Technology* 49(3), 246–255 (2005)
14. Menesatti, P., Antonucci, F., Pallottino, F., Rocuzzo, G., Allegra, M., Stagno, F., Intrigliolo, F.: Estimation of plant nutritional status by vis-nir spectrophotometric analysis on orange leaves (*Citrus sinensis* (L.) Osbeck cv tarocco). *Biosystems Engineering* 105(4), 448–454 (2010)
15. Mertens, K., Vaesen, I., Loffel, J., Kempfs, B., Kamers, B., Perianu, C., Zoons, J., Darius, P., Decuyper, E., De Baerdemaeker, J., De Ketelaere, B.: The transmission color value: A novel egg quality measure for recording shell color used for monitoring the stress and health status of a brown layer flock. *Poult. Sci.* 89(3), 609–617 (2010)
16. Pap, K., Žiljak, I., Žiljak Vujić, J.: Image reproduction for near infrared spectrum and the infrared design theory. *Journal of Imaging Science and Technology* 54(1), 010502 (2010)
17. Shin, H., Reyes, N.H.: Variable colour depth look-up table based on fuzzy colour processing. In: Köppen, M., Kasabov, N., Coghill, G. (eds.) *ICONIP 2008*. LNCS, vol. 5506, pp. 1071–1078. Springer, Heidelberg (2009)
18. Shin, H.: Finding near optimum colour classifiers: Genetic algorithm-assisted fuzzy colour contrast fusion using variable colour depth. Master's thesis, Massey University (2009)
19. Reyes, N.H., Dadios, P.E.: Dynamic color object recognition using fuzzy logic. *Journal of Advanced Computational Intelligence and Intelligent Informatics* 8, 29–38 (2004)
20. Thomas, P., Stonier, R., Wolfs, P.: Robustness of color detection for robot soccer. In: *Proceedings of the Seventh International Conference on Control, Automation, Robotics and Vision*, pp. 1245–1249 (2002)
21. Shin, H., Reyes, N.: Finding near optimum colour classifiers: genetic algorithm-assisted fuzzy colour contrast fusion using variable colour depth. *Memetic Computing Journal*, 1–18 (2009)

An Adaptive Bidding Strategy for Combinatorial Auction-Based Resource Allocation in Dynamic Markets

Xin Sui^{1,*} and Ho-fung Leung²

¹ Department of Computer Science
University of Toronto
Toronto, Ontario M5S 3G4, Canada
xsui@cs.toronto.edu

² Department of Computer Science and Engineering
The Chinese University of Hong Kong
Sha Tin, Hong Kong, China
lhf@cuhk.edu.hk

Abstract. Combinatorial auction, where bidders can bid on bundles of items, has been the subject of increasing interest in recent years. Although much research work has been conducted on combinatorial auctions, most has focused on the winner determination problem. A largely unexplored area of research in combinatorial auctions is the design of bidding strategies. In this paper, we propose a new adaptive bidding strategy for combinatorial auction-based resource allocation problem in dynamic markets. A bidder adopting this strategy can adjust his profit margin constantly according to his bidding history, thus perceiving and responding to the dynamic market in a timely manner. Experiment results show that agents adaptive bidding strategy perform very well, even without any prior knowledge about the market.

Keywords: combinatorial auctions, resource allocation, adaptive strategies.

1 Introduction

The use of computing power provided by centralized and distributed infrastructures is of increasing interest of research in recent years in computer science. Internet is an example of such infrastructures where different users (people, software agents) can use the provided computational resources to perform their own tasks [5]. The resource allocation problem, that is, how to distribute resources among a group of users, receives much attention and becomes an important issue. Internet auction is a natural choice to solve this kind of resource allocation problems, because it allocates resources to the bidders who value them most and achieves an efficient allocation of resources from the view of economics [2].

* The research was done when the first author was with Department of Computer Science and Engineering, The Chinese University of Hong Kong.

Combinatorial auctions, where bidders are allowed to put bids on bundles of items, receive much attention from researchers in both computer science and economics [4]. Combinatorial auctions can lead to more economical allocations of resources than conventional single-item auctions when bidders have complementarities (substitutability) among them. Such an advantage can lead to an improvement of efficiency, which has also been demonstrated in airport landing allocation and transportation exchanges [9][11].

There has been a surge of research interests in combinatorial auctions in the last decade. The two most widely studied problems are winner determination and auction design. Winner determination problem is about finding the optimal allocation of resources among a group of bidders. This optimization problem has been proved to be *NP*-hard in general case [10], and much work has been conducted for solving it, including finding both optimal solutions and approximate solutions [12][6][16]. Combinatorial auction design involves the investigation of the design of different auction protocols for combinatorial auctions, such as single-round versus multi-round, open-cry versus sealed-bid, and the use of various bidding rules [8][3].

A largely unexplored area of research in combinatorial auctions is the investigation of bidding strategies. As combinatorial auctions are always incorporated with the first-price sealed bid auction protocol in many applications [3], we are especially interested in bidding strategies in this kind of auctions. In this paper, we consider a scenario where first-price sealed-bid combinatorial auctions are employed to distribute computational resources among a group of users, and propose a novel adaptive bidding strategy. A bidder adopting this kind of strategy adjusts his profit margin from time to time according to his bidding history, thus perceiving and responding to the dynamic markets. Experiment results show that bidders with the adaptive strategy obtain high utilities in different dynamic markets.

This paper is structured as follows. Section 2 presents related work. Section 3 presents the combinatorial auction model. Section 4 describes the adaptive bidding strategy. Section 5 shows simulation results. Finally Section 6 concludes this paper and highlights some future work.

2 Related Work

Resource allocation problem is an important issue in the area of computer science. In recent years, a lot of work has been conducted for solutions, among which centralized mechanisms and distributed mechanisms are two main approaches. In centralized mechanisms, there is a resource manager that decides how to allocate resources among a group of resource consumers, while in distributed mechanisms, consumers coordinate implicitly or explicitly with one another to reach an agreement of the allocation of resources.

Schwind *et al.* [14] attempt to solve the computational resource allocation problem using multi-round combinatorial auctions. They study the situation where bidders spend virtual currencies, which are obtained by selling unused resources, to get accesses to computational resources needed for accomplishing their own tasks. They propose bidding strategies for two types of bidders: 1) impatient bidders, who benefit from the instantaneously use of resources and 2) quantity maximizing bidders, who

require high resource capacities but have weak preferences regarding the timing. Experiment results show that for the first type of bidders, it is better to bid aggressively to get fast accesses to resources, while the second type of bidders had better bid low prices and keep on waiting for resources.

Sui and Leung [15] also try to employ multi-round combinatorial auctions to distribute computational resources among a group of users. They propose an adaptive bidding strategy for bidders in static markets where the ratios of supplies to demands of resources are kept constant during the whole process of the auction. A bidder adopting this kind of strategy can adjust his profit margin from time to time according to his bidding history, and finally adapts to the current market environment even without any prior knowledge about the market. Through simulations, they show that a bidder using the adaptive strategy outperforms others using other strategies, and receive high utilities when compared with optimal strategies in several static markets.

Galstyan *et al.* [7] study the resource allocation problem with a changing capacity. In their work, each user uses a set of lookup tables to decide which resource to choose and use a simple reinforcement learning scheme to record the accuracy of these tables. A lookup table guides the user's decision based on the neighbours' actions at previous time steps. At the end of each time step, each user assesses the performance of his lookup tables by increasing or reducing a point of score, depending on whether it has correctly predicted a winning choice. Experiment results show that users can adapt effectively to changing capacities in dynamic markets.

Schlegel and Kowalczyk [13] propose a self-organizing distributed resource allocation algorithm. They study the case where multiple servers are providing identical resources with a changing capacity to a group of resource consumers. For each consumer, a decision on which server his task is executed is made independently according to the predictor that is randomly chosen from a set of predictors. The probability that a certain predictor is chosen is increased if a correct prediction is made, or is decreased when a wrong prediction is made. Experiment results show that the bidder using the proposed approach can adapt to dynamic markets and their collaborative behaviour achieves a good effect of resource load balancing.

3 Model Description

A combinatorial auction for the computational resource allocation problem is as follows. There are m different types of resources provided by a resource provider, *e.g.*, a server or a grid platform, to a group of n users. For each type $j \in \{1, 2, \dots, m\}$ of resource, there is a capacity c_j that denotes the number of units currently available. The value of c_j generally varies over time. Such a market is called a *dynamic market*.

Each user needs certain resources to perform his task, and the maximum number of units of type j resources that a user can request for is m_j . Each user $i \in \{1, 2, \dots, n\}$ submits a scaled-bid $b_i = (R, p_i(R))$ to the resource provider, where $R \in \{r_1, r_2, \dots, r_m\}$, $r_j \leq m_j$, $1 \leq j \leq m$, contains the number of different resources that he requests for, and $p_i(R)$ is a positive number denoting the price he will pay for getting R . After receiving bids from all users, the resource provider solves the winner

determination problem, that is, to find the allocation maximizing his revenue with the constraint that for each type of resource j , the total number of units allocated does not exceed its capacity c_j . Winning users will pay their bidding prices to get accesses to the resources they bid for, perform their tasks, and then return the resources to the resource provider. We refer to the process from the beginning of bid submission to the end of resource return as a *round* of a combinatorial auction. Because such computational resources are reusable, the combinatorial auction can be repeated for multiple rounds before it is closed by the resource provider.

Before we describe our adaptive bidding strategy, we list some assumptions used in this paper. First, we assume that the information available to each bidder is his own bidding information in the previous rounds *only*, e.g., his previous bids and bidding results, and any information about other bidders, such as other bidders' previous bids and bidding results, is not accessible. Second, each bidder only submits one bid per round, which is determined by the resource bundle he needs for his current task. Hence a bidder who won in the previous round will submit a (generally) new bid, while those who lost continue to submit the lost bids. However, a bid will be given up after having been submitted for τ consecutive rounds and a new bid with a new resource bundle will be submitted. This simulates the fact that a bidder has a limited patience on waiting.

4 Adaptive Bidding Strategy

As described in the above section, each winner needs to pay the price he has bid to get the resources, and each loser pays nothing. For a bid $(R, p_i(R))$, each bidder i has its own valuation $v_i(R)$ of bundle R . A rational bidder will use a $p_i(R)$ that is less than $v_i(R)$, otherwise he will get a negative utility when winning. That is, $p_i(R) = (1 - pm_i) \times v_i(R)$, where $pm_i \in [0, 1]$ is known as bidder i 's *profit margin for the bid* $(R, p_i(R))$. The utility of bidder i is hence:

$$u_i(R) = \begin{cases} pm_i \times v_i(R) & i \text{ wins} \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

Now, a bidder faces a dilemma. Bidding with a low profit margin generally increases his winning opportunity, but decreases his winning utility at the same time. The opposite is also true. If a bidder somehow is able to get some prior knowledge about the market environment, e.g., the number of bidders competing for resources, he *might* probably be able to make use of such information to decide wisely. For example, a bidder who knows that there are more supplies than demands should tend to use a higher profit margin when bidding. However, as we assume in this paper, in an open and dynamic environment, such information is usually inaccessible. Furthermore, the market environment *can* vary from time to time. How to design an adaptive bidding strategy that can help the bidder perceive the environment and respond to the market in a timely manner, even with limited information, is a challenge.

4.1 Basic Concepts

Before we introduce the adaptive strategy, some basic concepts are defined.

Definition 1. A **bidding record** of a bid $b_i = (R_b, p_i(R_b))$ of bidder i is a tuple $br_b = (R_b, v_i(R_b), pm_i, w_b, res_b)$, where R_b is the resource bundle required in this bid, $v_i(R_b)$ is bidder i 's valuation for R_b , pm_i is the profit margin used by bidder i in this bid, $w_b \in [0, \tau]$ is the number of rounds the bidder i keeps on bidding with the bid before the bid is accepted ($res_b = 1$) or given up ($res_b = 0$).

Definition 2. The **anticipated utility** $u_{br}(br_b)$ of a bidding record br_b is

$$u_{br}(br_b) = pm_i \times (res_b / (res_b + w_b)) \tag{2}$$

Hence, the minimum value of $u_{br}(br_b)$ is 0, where the bid is rejected and $w_b = \tau$; while its maximum value is pm_i , if the bid is accepted in the first round.

Definition 3. The **bidding history** of a bidder is the sequence of the most recent λ bidding records.

Therefore, every time when a bid is accepted or given up, the oldest bidding record is removed from the bidding history and the new bidding record is appended to the bidding history. We will use bh^* to denote the current bidding history. We define the *age* of a bidding record as follows.

Definition 4. The **age of a bidding record** br_b in bh^* is the number of times bh^* is updated after br_b is appended to bh^* .

For any bidding record in bh^* , its age is always between 0 and $\lambda - 1$. No bidding history contains any bidding record of age older than λ .

In a dynamic market, information contained in newer bidding records is more valuable than that in an older bidding record. We define the weights of each bidding record as follows:

Definition 5. A **weight function on the age of bidding records** is a decreasing function $f_w : \{0, 1, \dots, \lambda\} \rightarrow [0, 1]$ that maps the ages of bidding records to the importance of the information contained in the bidding records.

The newer a bidding record is, the better it can reflect the current market environment, and the higher weight it is given.¹

Definition 6. The **anticipated utility** $u_{bh}(bh^*)$ of the **bidding history** bh^* is the weighted average anticipated utilities of bidding records in bh^* :

$$u_{bh}(bh^*) = \sum_{br_b \in bh^*} f_w(br_b) \times u_{br}(br_b) / \sum_{br_b \in bh^*} f_w(br_b) \tag{3}$$

Based on definition 6, we give two notations of $u_{bh}(bh^* \upharpoonright_{\geq \sigma})$ and $u_{bh}(bh^* \upharpoonright_{\leq \sigma})$, which denote the weighted average anticipated utilities of bidding records in bh^* , whose profit margins are not less than and not more than σ , respectively.

¹ A sample weight function is $f_w(a) = 1 - a/\lambda$.

$$u_{bh}(bh^* \uparrow_{\geq \sigma}) = \frac{\sum_{br_b \in bh^* \uparrow_{\geq \sigma} (R_b, v_i(R_b), pm_i, w_b, res_b), pm_i \geq \sigma} f_w(br_b) \times u_{br}(br_b)}{\sum_{br_b \in bh^* \uparrow_{\geq \sigma} (R_b, v_i(R_b), pm_i, w_b, res_b), pm_i \geq \sigma} f_w(br_b)} \quad (4)$$

$$u_{bh}(bh^* \downarrow_{\leq \sigma}) = \frac{\sum_{br_b \in bh^* \downarrow_{\leq \sigma} (R_b, v_i(R_b), pm_i, w_b, res_b), pm_i \leq \sigma} f_w(br_b) \times u_{br}(br_b)}{\sum_{br_b \in bh^* \downarrow_{\leq \sigma} (R_b, v_i(R_b), pm_i, w_b, res_b), pm_i \leq \sigma} f_w(br_b)} \quad (5)$$

4.2 An Adaptive Strategy

The basic idea of the adaptive strategy is that a bidder should continuously review and revise his profit margin in use. This process is called an *adaptation* of the profit margin, through which a bidder aims to be able to dynamically maintain a good profit margin in response to the ever-changing market environment.

Algorithm 1. Adaptive Strategy

```

1:  pm = η, step = θ, δ = 1 and u' = 0 .
2:  while auction does not finish do
3:    Use pm to bid for the subsequent rounds
4:    if a new bidding record br_b is formed then
5:      Update bh^* and compute u_{br}(br_b).
6:      u = u_{br}(br_b) and pm' = pm.
7:      if CheckStepDecrease() = true then
8:        Decrease step by γ;
9:      else if CheckStepIncrease() = true then
10:       Increase step by γ;
11:     end if
12:     if u = 0 AND u' = 0 then
13:       pm = pm - step
14:     else if u ≠ 0 AND u' ≠ 0 then
15:       if u < u' then pm = pm - δ × step else if u ≥ u' then pm = pm + δ × step end if
16:     else if u = 0 OR u' = 0 then
17:       Compute u_{bh}(bh^* \uparrow_{\geq \sigma}) and u_{bh}(bh^* \downarrow_{\leq \sigma})
18:       if u_{bh}(bh^* \uparrow_{\geq \sigma}) < u_{bh}(bh^* \downarrow_{\leq \sigma}) then pm = pm - step else pm = pm + step end if
19:     end if
20:     if pm > pm' then δ = 1 else if pm < pm' then δ = -1 end if
21:     u' = u
22:   end if
23: end while

```

We use a 0-1 variable δ to indicate the direction of an adjustment of the profit margin: if $\delta = 1$, then the adjustment is positive, otherwise negative. In addition, we use u and u' to denote the anticipated utilities of the most and the second most recent bidding records. Finally, while pm denotes the current profit margin, pm' denotes the profit margin before the previous adjustment.

The adaptive strategy is illustrated in Algorithm 1. Function CheckStepDecrease (line 7) and CheckStepIncrease (line 9) check whether *step* needs to be increased or decreased. We note that if a small value is used for *step*, the adaptive strategy will need a long time to approach to the new optimal profit margin when market environment changes, but the profit margin generated by the adaptive strategy can be more

refined. On the other hand, if a large value is used, the profit margin can be adjusted more quickly when market environment changes, but the bidder might over adapt to changes. Therefore, we need to adjust the value of *step* dynamically during the process of the adaptation.

The adaptive strategy can be illustrated as follows. At first, *pm*, *step*, δ and u' are initialized. During the process of the auction, the bidder reviews the value of *pm* whenever a new bidding record is formed. To decide how to change *pm*, the bidder first updates the bidding history and computes the anticipated utility u of the latest bidding record (lines 5 to 6). In u and u' are both 0, the current profit margin is thought to be too high, and is then decreased by *step* (lines 12 to 13). If neither u nor u' is 0, but the previous adjustment of the profit margin (recorded by δ on line 19) has led to a decrease of the anticipated utility, an adjustment in the opposite direction will be made; otherwise, an adjustment in the current adjustment direction will be made (lines 14-15). Finally, if only one of u and u' is 0, it is not clear how the profit margin should be adjusted, because an anticipated utility of 0 may be caused by many reasons, e.g., a low valuation of the bundle, rather than a low profit margin used. In this case we rely on the bidding history: If $u_{bh}(bh^* l_{\geq \alpha}) < u_{bh}(bh^* l_{\leq \alpha})$, which means that decreasing the profit margin will obtain a higher average anticipated utility, the bidder will make a negative move, otherwise a positive move (lines 16-17).

Next, we will describe the CheckStepDecrease and CheckStepIncrease functions in details.

Algorithm 2. Function: CheckStepDecrease

```

1: Compute  $mean = \frac{1}{\kappa} \sum_{i=1}^{\kappa} pmh^i$ .
2: for  $i = 1$  to  $\kappa$  do if  $|pmh^i - mean| \leq sh^i$  then  $\omega^i = 1$  else  $\omega^i = 0$  end if end for
3: return  $\sum_{i=1}^{\kappa} \omega^i \geq \varphi$  AND  $\omega^1 = 1$  AND  $pm \Rightarrow mean$  AND  $step > \alpha$ 

```

4.2.1 Function I: CheckStepDecrease

As mentioned above, the value of *step* should be adjusted from time to time with the hope that the profit margin revised by the adaptive strategy will be more suitable to the current market situation. The function CheckStepDecrease determines when to decrease the value of *step*. We now define a number of notations as follows.

Definition 7. The **profit margin history** *pmh* is a sequence of κ profit margins used in the most recent κ bidding records.

Definition 8. The **step history** *sh* is a sequence of κ values used as *step* in the most recent κ profit margin updates.

We use the notation $pm \Rightarrow \pi$ to denote 1) $pm < \pi$ and the next adjustment for *pm* is positive; or 2) $pm > \pi$ and the next adjustment for *pm* is negative.

The function CheckStepDecrease is given in Algorithm 2. We first compute the mean value of the elements in *pmh* (line 1), then for each element in *pmh* we check if it is close enough to the mean (line 2). To decrease *step*, we need several conditions to be satisfied (line 3). First, if a significant number of elements in *pmh* are fluctuating

around the mean, then we regard the mean as an approximation of the optimal profit margin in the current market environment. Second, if the *last* element in *pmh* is close enough to the mean, and $pm \Rightarrow mean$, then the optimal profit margin can be approached with higher accuracy if *step* is decreased. Finally, *step* cannot be too small (smaller than α). If all these conditions hold, the function returns true.

Algorithm 3. Function: CheckStepIncrease

```

1:  negMove = 0, posMove = 0.
2:  for  $i = 1$  to  $\kappa - 1$  do
3:      if  $pmh^i < pmh^{i+1}$  then negMove++ else if  $pmh^i > pmh^{i+1}$  then posMove++ end if
4:  end for
5:  return ( negMove  $\geq \chi$  AND  $\delta = -1$  AND  $step < \beta$  )
      OR ( posMove  $\geq \chi$  AND  $\delta = 1$  AND  $step < \beta$  )

```

4.2.2 Function II: CheckStepIncrease

The function CheckStepIncrease given in Algorithm 3 determines when to increase *step*. We count the numbers of positive and negative moves made in the profit margin history, respectively (lines 2-4). Note that it never happens that $pmh^i = pmh^{i+1}$ because by Algorithm 1, a positive or negative move is always made when a new bidding record is formed. If there are many negative moves in the profit margin history (more than χ) and the next move of the profit margin is negative (first part of line 5), we believe that the market environment is becoming more competitive for resource consumers and the bidder should increase the value of *step* to adapt to the new market quickly. Similarly, *step* should also be increased when there are many positive moves in the profit margin history (more than χ) and the next move of the profit margin is positive (second part of line 5). In either case, if the threshold value β to stop increasing *step* is not reached, the function will return true.

5 Experiment Evaluation

To evaluate the performance of the adaptive strategy, we conduct two sets of experiments. In the first set of experiments, we try to identify the optimal profit margins in different market environments. In the second set of experiments, we show that the adaptive strategy outperforms the random strategy and its performance is very close to an oracle strategy that makes use of market information that is assumed to be inaccessible. In addition, we also illustrate the typical adaptation processes of the profit margin in dynamic markets in the second set of experiments.

5.1 First Set of Experiments: Estimation of the Optimal Profit Margin

5.1.1 Experiment Setup

In the first set of experiments, our aim is to find the best profit margins in markets with different supply/demand ratios. This is done by first testing a set of fixed strategies that use fixed profit margins throughout the whole process of the auction. We use 19 different fixed strategies with profit margins $pm_1 = 0.05$, $pm_1 = 0.10$, $pm_3 = 0.15$, ..., $pm_{19} = 0.95$.

We find the best fixed strategy for a particular type of market as follows. For each fixed strategy, we repeat the combinatorial auction for 100 runs, with each run consisting of 500 rounds of combinatorial auctions. Follow-

Table 1. Parameters used in the experiments

<i>Parameter</i>	<i>Value Used</i>	<i>Description</i>
τ	3	Maximum lost round
λ	5	Length of a bidding history
η	0.05	Initial value of <i>pm</i>
θ	0.1	Initial value of <i>step</i>
γ	2	Amount of decrease or increase for <i>step</i>
κ	10	Length of a profit margin history
φ	6	See Algorithm 2
α	0.01	Threshold to stop decreasing <i>step</i>
χ	7	See Algorithm 3
β	0.1	Threshold to stop increasing <i>step</i>

ing the previous work [1][14], in each run, we have one testing bidder using that fixed strategy while others are bidding with their true valuations. After 100 runs, the accumulated utilities obtained by the testing bidders using different fixed strategies are compared, and the best performing fixed strategy is identified.

The experiment settings are as follows. A group of $n = 60$ users compete for $m = 4$ types of resources provided by a resource provider. For each bidder, the numbers of units that he can request for different resources are integers randomly drawn from uniform distributions [0, 3], [0, 2], [0, 2] and [0, 1]. His valuations for single unit of different resources are real numbers randomly drawn from uniform distributions [3, 6], [4, 8], [4, 8] and [6, 10]. For a resource bundle R which contains more than one type of resources, a synergy seed, $syn(R)$, is randomly drawn from a uniform distribution [-0.2, 0.2], and his valuation for that bundle is the product of sum valuations of individual resources and $1 + syn(R)$. Positive synergy seed means complementarities among resources and negative synergy seed means substitutability among them.

Table 1 summarises the parameters used in experiments.

5.1.2 Experiment Results and Analysis

The simulation results of the first set of experiments are shown in Fig. 1. Each curve represents a certain market environment, and the accumulated utilities of the testing bidders using 19 different fixed strategies are compared. We can see that the less competitive the market is for resource consumers, the higher the value of the best fixed profit margin will be. For example, in the market with a supply/demand ratio of 1.2:1, the best fixed profit margin is 0.95, while in the market with a supply/demand ratio of 0.5:1, the best fixed profit margin is 0.15. This agrees with our expectation that in a market less competitive for bidders, it is better to use a high profit margin, and vice versa.

Based on the results in Fig. 1, we use a regression method to interpolate these optimal values to approximate the optimal profit margins in different market environments. The result is shown in Fig. 2. The red dots are best performing fixed profit margins obtained from Fig. 1, and are regarded as sample points. We use a piecewise function $opt(rf)$ to fit these samples:

$$opt(rf) = \begin{cases} a \times b^{rf} + c & rf < \rho \\ d & rf \geq \rho \end{cases} \quad (6)$$

The result of the regression is that $a = 0.0001334$, $b = 2561.574$, $c = 0.1645$, $d = 0.05$ and $\rho = 1.106062$, which is shown as the blue line in Fig. 2. We can see that the blue line fits our samples very well, and when talking about a certain type of market denoted by rf , we will use the function value as the optimal profit margin.²

We can imagine a bidder who somehow has access to equation (6) and the current rf can always make a very good decision on what profit margin to use. He is actually using a bidding strategy that is practically impossible. We shall refer to such a bidding strategy an *oracle strategy*, which will be used as a benchmarking strategy in the second set of experiments to evaluate the performance of the adaptive strategy.

5.2 Second Set of Experiments: Performance of the Adaptive Strategy

5.2.1 Experiment Setup

In this section, we compare the performance of the random strategy, the oracle strategy and the adaptive strategy. The random strategy is a strategy that a random profit margin is used for each bidding record. The oracle strategy is a strategy that the bidder is privileged and has complete knowledge of the current market environment, which is denoted by the value of rf , and always uses the best profit margin for the latest market given by equation (6) when bidding. Note that in equation (6), the maximum profit margin can be generated is 0.95. Therefore, we also set up an upper bound of 0.95 on the profit margins generated by both strategies.³

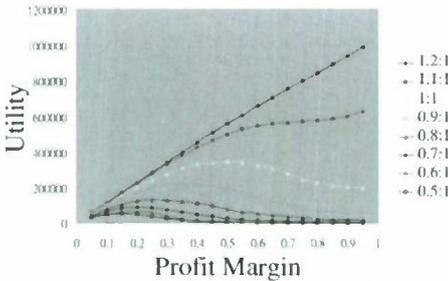


Fig. 1. Utilities of testing bidders using 19 different fixed strategies in markets with different supply/demand ratios

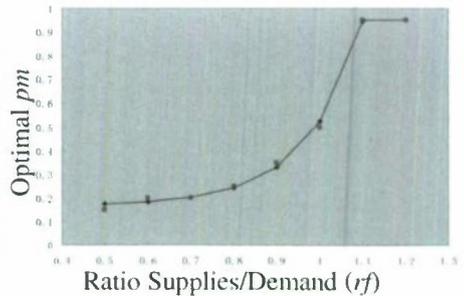


Fig. 2. Regression curve of the optimal profit margin

² Here, an exponential function is used as the left part of the regression curve, and actually, it does not matter too much if we use other functions. This is because in the second set of experiments, we never use equation (7) to estimate the optimal profit margin of the market whose rf falls out of $[0.5, 1.2]$, and the estimated optimal profit will not vary much if other fitting functions are used.

³ Actually, setting this upper bound does not affect the performance of the adaptive strategy. This is because without this constraint, when the optimal profit margin is a value infinitely close to 1, the profit margin generated by the adaptive strategy is also very close to 1, and the bidder using the adaptive strategy does not losing utility at all.

A dynamic market is one in which the value of rf changes by the time. We consider three types of dynamic markets, which are shown in the left column of Fig. 3. The first one is that the capacity factor rf changes in a linear pattern and keeps as constant alternatively, the second one is that rf changes in a linear pattern, and the last one is that rf changes in a cosine pattern. A run is now composed of 900 rounds. Other settings are the same as those in section 5.1.1.

5.2.2 Experiment Results and Analysis

The middle column in Fig. 3 shows the simulation results of the second set of experiments. The utilities obtained by the bidders using the adaptive strategy (AS), the oracle strategy (IS) and the random strategy (RS) are compared. We can see that the bidders using the adaptive strategy perform fairly well: they outperform the random strategy and obtain good utilities compared to the oracle strategy in all dynamic markets.

We also show in the right column of Fig. 3 some typical adaptation processes of the profit margin in a single run in different dynamic markets. The red lines indicate the optimal profit margins given by equation (6) and the blue lines show the profit margins used by the adaptive strategy. We can see that the profit margin used by the adaptive strategy is close to the optimal profit margins given by equation (6). This means that the bidder using the adaptive strategy is capable of adapting to different dynamic markets. In addition, the adaptation is timely, even when the optimal profit margin changes sharply, e.g., the change of the profit margin shown in Fig. 3.c.

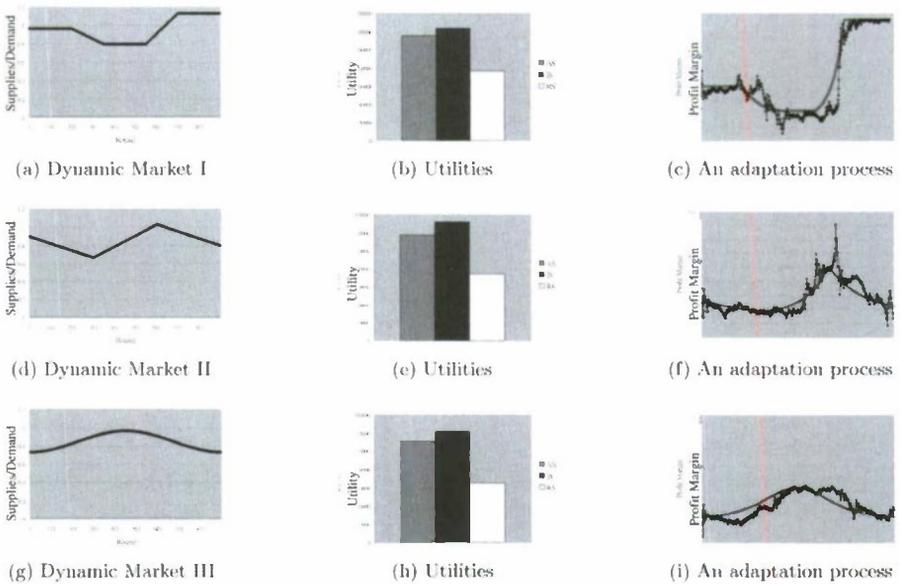


Fig. 3. Performance and Adaptation Process of the Adaptive Strategy in Different Dynamic Markets

6 Conclusions and Future Work

In this paper, we propose a new adaptive bidding strategy for combinatorial auctions-based resource allocation problem in dynamic markets. The bidder adopting this strategy can adjust his profit margin from time to time according to his bidding history and thus perceive and respond to the changing market environment. Through simulations, we show that 1) the adaptive strategy performs fairly well compared to the random strategy and the oracle strategy in different dynamic markets. 2) the bidder using the adaptive strategy can obtain high utilities, even without any prior knowledge about the market. 3) the bidder using the adaptive strategy is capable of adapting in dynamic markets and responds in a timely manner.

There are some points for our future work. First, we assume in this paper that the computational resources to be auctioned are reusable. There are also many applications where the auctioned resources are non-reusable. Next step, we are going to study the adaptive behaviour in such type of auctions. Second, in this paper, we only consider the situation where the supplies of resources vary gradually over time, and the effectiveness of the adaptive strategy in markets with abrupt changes is not yet known. In the future, we intend to study the market, where both the number of bidders and the capacities of resources can change. Finally, from the simulation results we can see that although the adaptive strategy performs well, there is still space for improvement. We are going to explore the influences of different parameters on the performance of the adaptive strategy.

References

1. An, N., Elmaghraby, W., Keskinocak, P.: Bidding strategies and their impact on revenues in combinatorial auctions. *J. Rev. Pricing Manage.* 3(4), 337–357 (2005)
2. Clearwater, S.: *Market-Based Control: A Paradigm for Distributed Resource Allocation*. World Scientific, Singapore (1996)
3. Cramton, P., Shoham, Y., Steinberg, R.: *Combinatorial Auctions*. MIT Press, Cambridge (2006)
4. de Vries, S., Vohra, R.: Combinatorial Auctions: A Survey. *INFORMS. J. Comput.* 15(3), 284–309 (2003)
5. Foster, I.: The Anatomy of the Grid: Enabling Scalable Virtual Organizations. In: Sakellariou, R., Keane, J.A., Gurd, J.R., Freeman, L. (eds.) *Euro-Par 2001*. LNCS, vol. 2150, p. 1. Springer, Heidelberg (2001)
6. Fujishima, Y., Leyton-Brown, K., Shoham, Y.: Taming the computational complexity of combinatorial auctions: Optimal and approximate approaches. In: *IJCAI*, pp. 548–553 (1999)
7. Galstyan, A., Kolar, S., Lerman, K.: Resource allocation games with changing resource capacities. In: *AAMAS*, pp. 145–152 (2003)
8. Parkes, D., Ungar, L.: Iterative combinatorial auctions: Theory and practice. In: *AAAI*, pp. 74–81 (2000)
9. Rassenti, S., Smith, V., Bulfin, R.: A combinatorial auction mechanism for airport time slot allocation. *Be. J. of Econ.* 13(2), 402–417 (1982)
10. Rothkopf, M., Pekec, A., Harstad, R.: Computationally Manageable Combinatorial Auctions. *Manage. Sci.* 44(8), 1131–1147 (1998)

11. Sandholm, T.: An implementation of the contract net protocol based on marginal cost calculations. In: *AAAI*, pp. 256–262 (1993)
12. Sandholm, T., Suri, S., Gilpin, A., Levine, D.: CABOB: A Fast Optimal Algorithm for Winner Determination in Combinatorial Auctions. *Manage. Sci.* 51(3), 374–390 (2005)
13. Schlegel, T., Kowalczyk, R.: Towards self-organising agent-based resource allocation in a multi-server environment. In: *AAMAS*, pp. 1–8. ACM, New York (2007)
14. Schwind, M., Stockheim, T., Gujo, O.: Agents' Bidding Strategies in a Combinatorial Auction Controlled Grid Environment. In: *TADA/AMEC*, pp. 149–163 (2006)
15. Sui, X., Leung, H.F.: An Adaptive Bidding Strategy in Multi-Round Combinatorial Auctions for Resource Allocation. In: *ICTAI*. IEEE Computer Society, Los Alamitos (2008)
16. Zurel, E., Nisan, N.: An efficient approximate allocation algorithm for combinatorial auctions. In: *ACM EC*, pp. 125–136 (2001)

Online Self-reorganizing Neuro-fuzzy Reasoning in Interval-Forecasting for Financial Time-Series

Javan Tan and Chai Quek

C²I, School of Computer Engineering, Nanyang Technological University,
N4, #B1a-02, Nanyang Avenue, Singapore 639798
{y060020,ashcquek}@ntu.edu.sg
<http://www.c2i.ntu.edu.sg>

Abstract. “The only thing constant is change.”—Ray Kroc (Founder of McDonald’s). Self-organizing neuro-fuzzy machines are maturing in their online learning process for time-invariant conditions. To, however, maximize the operative value of these self-organizing approaches for online-reasoning, such self-sustaining mechanisms must embed capabilities that aid the reorganizing of knowledge structures in real-time dynamic environments. Also, neuro-fuzzy machines are well-regarded as approximate reasoning tools because of their strong tolerance to imprecision and handling of uncertainty. Recently, Tan and Quek (2010) discussed an online self-reorganizing neuro-fuzzy approach called SeroFAM for financial time-series forecasting. The approach is based on the BCM theory of neurological learning via metaplasticity principles (Bienenstock et al., 1982), which addresses the stability limitations imposed by the monotonic behavior in Hebbian theory for online learning (Rochester et al., 1956). In this paper, we examine an adapted version called iSeroFAM for interval-forecasting of financial time-series that follows a computational efficient approach adapted from Lalla et al. (2008) and Carlsson and Fuller (2001). An experimental proof-of-concept is presented for interval-forecasting of 80 years of Dow Jones Industrial Average Index, and the preliminary findings are encouraging.

Keywords: neuro-fuzzy, fuzzy associative learning, online-learning, online-reasoning, self-organizing, self-reorganizing, evolving, time-variant, time-varying, BCM, bienenstock cooper mmuro, sliding threshold, synaptic plasticity, meta-plasticity, dissociative, anti-hebbian, interval-forecasting.

1 Introduction

In soft-computing sciences, online hybrids of neuro-fuzzy computing such as SAFIS [19], SimpleTS [3], eTS [2], DENFIS [13], EFuNN [12], SOFNN [15], and DFNN [26], are gaining popularity as cost-effective tools that can exploit tolerance for imprecision [27]. Neuro-fuzzy departure from the precision arts provide natural leeway for uncertainty and vagueness [23], which are typically unavoidable in many real-world forecasting problems. Extensive reviews of their properties are covered in [17,16,11].

Online neuro-fuzzy learning has been studied from two perspectives: 1) *time-invariant*, or 2) *time-variant*, depending upon the characterization of their underlying system dynamics, and the duration of the temporal-space involved. [22] discusses how for time-invariant problems with little or no temporal variation, neuro-fuzzy machines can effectively *self-organize* to learn out a final structure, where all data should be experienced with equal emphasis. If applied under time-variant conditions, such a self-organizer would actually average out the effects of the time-variance to obtain a middle-ground solution that would be a structural mix of obsolete and new information. Over time, the structure carries increasingly redundant information, and impairs the currency of results.

To manage more complex time-variant datasets that exhibit regime shifting properties, neuro-fuzzy machines need to continuously *self-reorganize* their internal structures to attune towards these pattern shifts in evolving data streams. This strong distinction in learning objectives is vital for modeling decision environments with changing characteristics, and has been widely discussed in advanced signal processing [8,9]. Generally, this necessitates increasing bias on recent data to identify persistent patterns, relative to transient ones. To account for information decay, forgetting factors [10], exponential gain functions [24] or adaptive training schemas [21] can be used to increase emphasis on more recent data experiences ([11] pp. 222). As such, Tan and Quek [22] described a tailored focus towards online-reasoning rather than online-learning, as self-reorganizing machines would generally serve transient reasoning purposes. Using self-reorganizing approaches to handle time-variance can be especially relevant in financial time-series forecasting, as shown in [22] and later in Section 3.

The second consideration for review is to enable interval-forecasting through a self-reorganizing approach. For many forecasting research, the idea is to train a model that can determine an accurate prediction for comparison to derive the lowest mean squared error and highest correlation scores. From a technical point of view, this approach makes sense. Accurate point-based forecasting is important, and is a valuable indicator for explicitly highlighting best fit trends. However, in truth, it is difficult to transfer the ownership of decision risk onto these forecasting computational models. There is no certainty in forecasting. Computational forecasting is about managing uncertainties and improving the decision making process, but ultimately they are only decision support tools. For this, we are concerned with prediction of an interval-forecast that can align neuro-fuzzy reasoning to the concept of volatility risk and uncertainty.

This paper proposes iSeroFAM, a modified interpretation of SeroFAM [22], which is computationally-based on the BCM-theory of meta-plasticity for online self-reorganizing fuzzy-associative learning. Here, the objective is to realize interval-forecasting capabilities as conceptualized by Carlsson and Fullér [6]. Section 2 provides a high-level overview of the learning approach. The paper focuses on Section 3 that examines the experimental proof-of-concepts for self-reorganizing interval-forecasting using iSeroFAM.

2 iSeroFAM: Self-reorganizing Fuzzy Associative Machine with Interval-Forecasting

The proposed iSeroFAM is an extension of SeroFAM [22], as means to exploit its self-reorganizing capabilities for interval-forecasting. It is an online neural connectionist construct with five neuronal-layers (see topology in Fig. 1). The actual input and output signals at any time t are given as crisp vectors $X(t) = [x_1, \dots, x_i, \dots, x_n]^T$ and $Y(t) = [y_1, \dots, y_j, \dots, y_m]^T$, and the symbols used in Fig. 1 denote: n as no. of input features; m as no. of output features; P_i as no. of membership functions (MFs) for x_i ; Q_j as no. of membership functions (MFs) for y_j ; L as no. of fuzzy premise nodes; $I^{(i)}$ as the input sensor for x_i , where $1 \leq i \leq n$; $IL_{p_i}^{(i)}$ as the p_i th MF for x_i , where $1 \leq p_i \leq P_i$; A_l as the l th fuzzy premise node, where $1 \leq l \leq L$; R_{l,q_j} as the fuzzy rule link between A_l and $OL_{q_j}^{(j)}$; $OL_{q_j}^{(j)}$ as the q_j th MF for y_j , where $1 \leq q_j \leq Q_j$; and $O^{(j)}$ as the output actuator for y_j , where $1 \leq j \leq m$.

Similarly, iSeroFAM operates by interleaving reasoning (testing) and learning (training) events, and the BCM learning process is described in detail in [22]. In brief, a rate-based Hebbian modification \dot{m} for rule learning is given as equ. (1):

$$\dot{m} = x \cdot \phi_{Hebb} = x \cdot y \tag{1}$$

where x and y are the pre-synaptic and post-synaptic signals of a neuron. However, it is not biologically plausible [7], since it implies that the synaptic weights

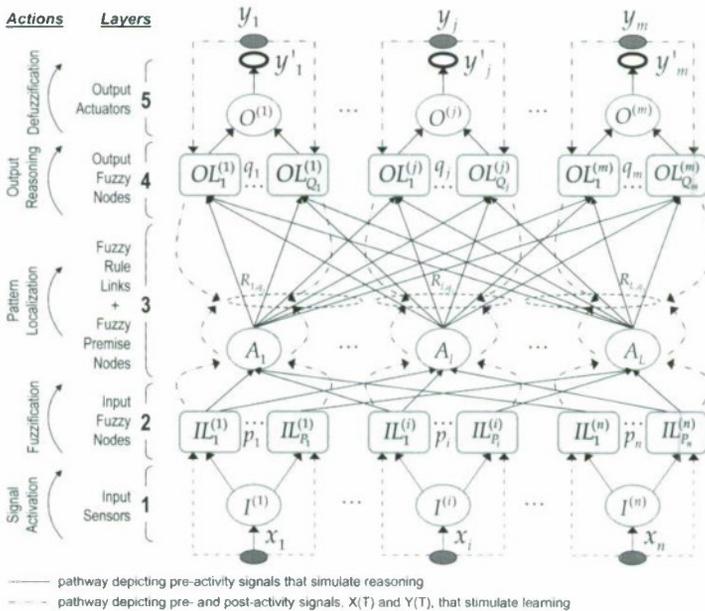


Fig. 1. Neuronal connections and layers for reasoning and learning events in iSeroFAM

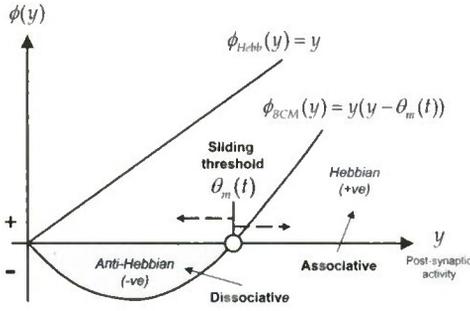


Fig. 2. Computational synaptic plasticity using the sliding threshold, θ_m

would reach arbitrarily large values over time. By acting independently at each synapse, Hebbian plasticity gains great power, but its *monotonic* behavior causes this stability problem [18], which makes it especially less useful for online computational learning.

The excitatory drive to a neuron has to be tightly regulated in an associative and dissociative manner (Hebbian and anti-Hebbian) to prevent saturation, otherwise information will be lost and *no selectivity will develop* [4]. Following which, [4] demonstrated that by floating a sliding threshold θ_m as a function of the averaged activity of the cell, it could overcome problems of runaway excitation and explain neural learning mechanisms. iSeroFAM applies a discrete computational form of the non-linear BCM activation function $\phi_{BCM} = y(y - \theta_m(t))$, as compared to the Hebbian activation function $\phi_{Hebb} = y$ as depicted in Fig. 2.

With reference to Fig. 1, the discrete update to the potential of the rule link $R_{l,q}$, between the premise node A_l and the output fuzzy node OL_q , at time t is can be written as eqn. (2):

$$\dot{P}_{l,q}(t) = \underbrace{\phi(\mu_{l,q}(t), \theta_{l,q}(t-1))f_{l,q}(t)}_{\text{Homosynaptic LTP (+ve)}} - \underbrace{\epsilon P_{l,q}(t-1)}_{\text{Heterosynaptic LTD}} \quad (2)$$

where: $P_{l,q}$ is the potential of $R_{l,q}$; $f_{l,q}$ is the pre-synaptic signal produced by A_l with a gaussian function; $\mu_{l,q}$ is the post-synaptic signal produced by y in OL_q with a gaussian function; $\theta_{l,q}$ is the sliding threshold based on $\overline{\mu_{l,q}^2}$; ϵ is the uniform decay given by $\epsilon = (1 - \lambda)$; and λ is the forgetting factor [22]. The first-half of eqn. (2) forms the basis for homosynaptic long-term potentiation (LTP) [5] and long-term depression (LTD) [20] depending on $sign(\phi)$, while the second-half explains exponential decay via heterosynaptic LTD [1].

2.1 Computation of Interval-Forecasts

The interval-forecast is conceptually based on the possibilistic variability measure described by Carlsson and Fullér [6]. To illustrate, first consider the output

$O^{(j)}$ having three fuzzy membership functions with centroids $c_1^{(j)}$, $c_2^{(j)}$, and $c_3^{(j)}$ as shown in Fig. 3. Assume input vectors **A** and **B** create two unique reasoning output spaces shown in Figs. 3(a) and 3(b) respectively. Note that both reasoning spaces will defuzzify to a same value, even though their possibilistic spread about the central value is differs.

Ceteris paribus, the reasoning for input vector **A** reflects a higher degree of confidence relative to input vector **B**. To quantify this variance, Lalla et al. (2008) [14] examined a centre-of-gravity (COG) variance that was computationally friendlier measurement than the mathematically derived possibilistic variance by Carlsson and Fullér [6]. Here, an mean-of-maxima (MoM) variance is implemented in Layer 5 of iSeroFAM shown in Fig. 1. The 5th activation relay $f^{V(j)}$ of iSeroFAM is modified to a MoM defuzzification as shown in eqn. (3), where $c_{q_j}^{(j)}$ is the centroid of the output fuzzy node, $OL_{q_j}^{(j)}$:

$$o^{V(j)} = f^{V(j)}(o_1^{IV(j)}, \dots, o_{q_j}^{IV(j)}, \dots, o_{Q_j}^{IV(j)}) = \frac{\sum_{j=1}^{Q_j} c_{q_j}^{(j)} \cdot o_{q_j}^{IV(j)}}{\sum_{j=1}^{Q_j} o_{q_j}^{IV(j)}} \tag{3}$$

In addition to the single crisp forecast represented by $o^{V(j)}$, the MoM variance, $\omega^{V(j)}$, is defined as a by-product computation at the defuzzification layer 5 based on eqn. (4):

$$\omega^{V(j)} = \frac{\sum_{j=1}^{Q_j} (c_{q_j}^{(j)} - o^{V(j)})^2 \cdot o_{q_j}^{IV(j)}}{\sum_{j=1}^{Q_j} o_{q_j}^{IV(j)}} \tag{4}$$

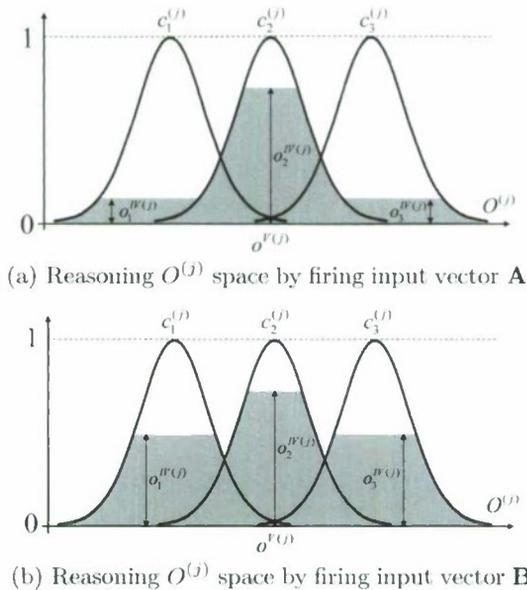


Fig. 3. Reasoning output spaces from iSeroFAM

Based on $o^{V(j)}$ and $\omega^{V(j)}$, an interval-forecast tuple with a lower-bound and upper-bound decision range can be computed as shown in eqn. (5), where k is the interval-multiplier that controls the range of the interval-forecast:

$$D^{V(j)} = [o^{V(j)} - k \cdot \omega^{V(j)}, o^{V(j)} + k \cdot \omega^{V(j)}] \tag{5}$$

3 Proof-of-Concept

The proof-of-concept experiments on iSeroFAM using real-world financial time-series data that is based on the Dow Jones Industrial Average (DJIA) index. About eighty years of daily index values was collected from the Yahoo! Finance website on the ticker symbol “^DJIA” for the period 2nd Jan 1930 to 31st Dec 2009, which provided 20,097 data-points for the experiment. Fig. 4 shows the movement of the index values with a time-variant exposure to the numerical range [41.22, 14164.53]. The main discussion will focus on the trajectory shifts of the index values that are especially rough after the 1980s, which is more noticeable from the increasing volatility in daily differences shown in the bottom half of Fig. 4. For the following experiments, the parameters as explained in [22] are: $G = 60$ days, $\rho = 0.8$, $b = 5.0$, and $z_{max} = 40$.

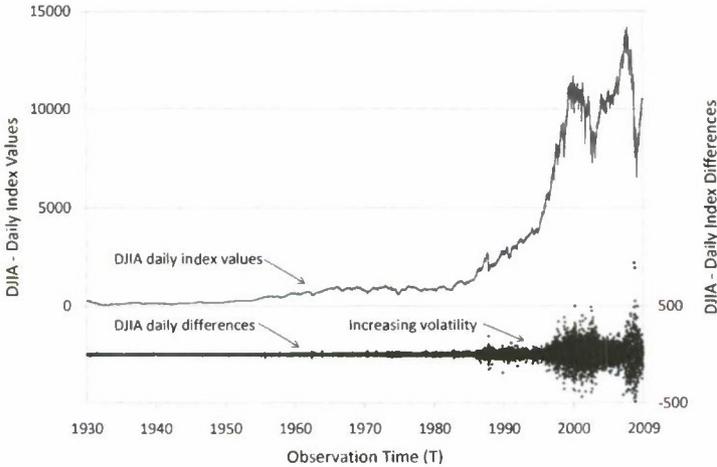


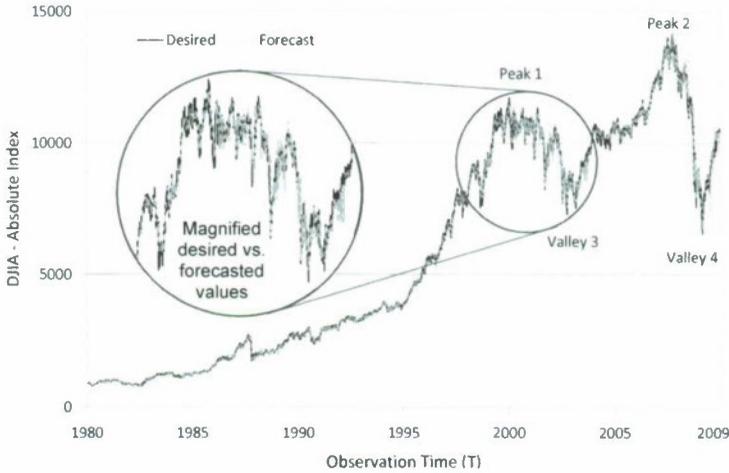
Fig. 4. Dow Jones Industrial Average daily index: 02 Jan 1930-31 Dec 2009 (80 years)

The analysis proceeds with an online precision-forecast of the DJIA index values using input vector, $X(T) = [m(T - 4), m(T - 3), m(T - 2), m(T - 1), m(T)]$ and output vector, $Y(T) = [m(T + 1)]$, where m is the absolute value of the DJIA index. Table 1 presents the experimental results. In this step-wise forecasting task, iSeroFAM performs with an overall NDEI = 0.0282 using an average of 19.4 rules with a PEARSON correlation of $R^2 = 0.999$. Based on availability

Table 1. Forecasting 80 years of DJIA market index

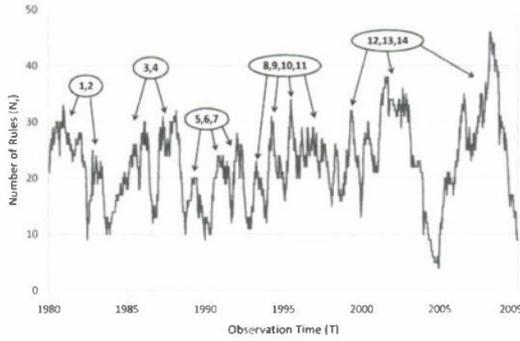
Model	Type	Ref.	Num. rules	NDEI	R^2
iSeroFAM [†]	Mamdani	-	19.4	0.0282	0.9996
EFuNN	Mamdani	[12]	91.6	0.1426	0.9917
DENFIS	T-S	[13]	5.0	0.0157	0.9999

[†]iSeroFAM is fully online self-reorganizing.

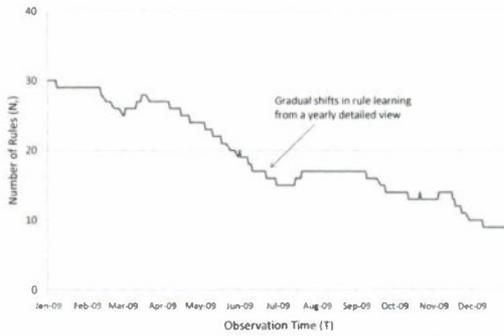
**Fig. 5.** Dow Jones Industrial Average $m(t + 1)$ forecasting results

constraints, this experiment was benchmarked against DENFIS and EFuNN for reference. From the tabulations, it is evident that iSeroFAM outperforms the Mamdani-based EFuNN [12] both in terms of accuracy and the number of rules generated. On the other hand, the Takagi-Sugeno-based (T-S) fuzzy-precision DENFIS model has a comparative advantage against iSeroFAM in terms of accuracy and number of rules used. However, the results have to be interpreted care. Although DENFIS and EFuNN are dynamic learning systems, they are not exactly online-reasoning. DENFIS normalizes data before learning, which indicates assumptions of prior knowledge of the upper and lower bound of the dataset, whereas only the rule nodes layer evolves in EFuNN [25]. On the other hand, online self-reorganizing models such as iSeroFAM are challenged without prior knowledge of the complete set of datapoints at any point in time.

Next, the analysis fast-forwards to the period of changes in the last 30 years, between 1980 to 2009. Fig. 5 provides the forecast plots of outputs, $m(T + 1)$ against the desired actuals. The forecasts from iSeroFAM noticeably follow through the trajectory shifts in the DJIA index, including the two peaks and two valleys occurring in years 2000, 2007, 2003 and 2009 respectively. During these 30 years of online learning, iSeroFAM performed at least 14, major reorganizations in the rule-base as can be observed from Fig. 6(a). During times of change, rules are quickly unlearned and new ones learnt to improve the currency of the knowledge representation. In real terms, iSeroFAM effectively re-learned its rule



(a) Self-reorganization of rule neurons over 30 years.



(b) Self-reorganization of rule neurons for Year 2009.

Fig. 6. Self-reorganization of associations in rule neurons during learning process

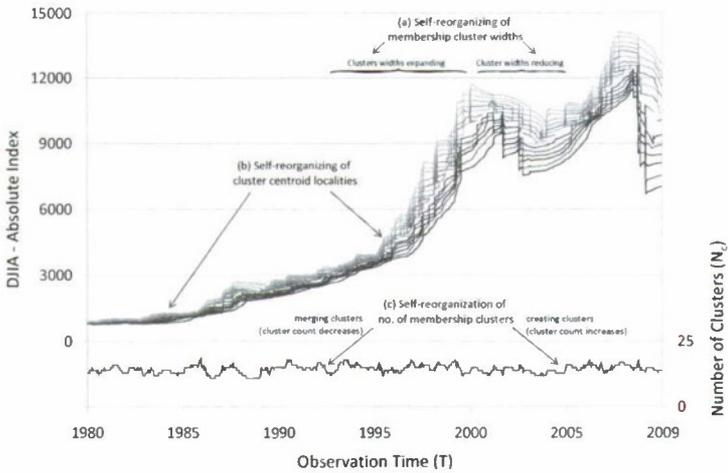
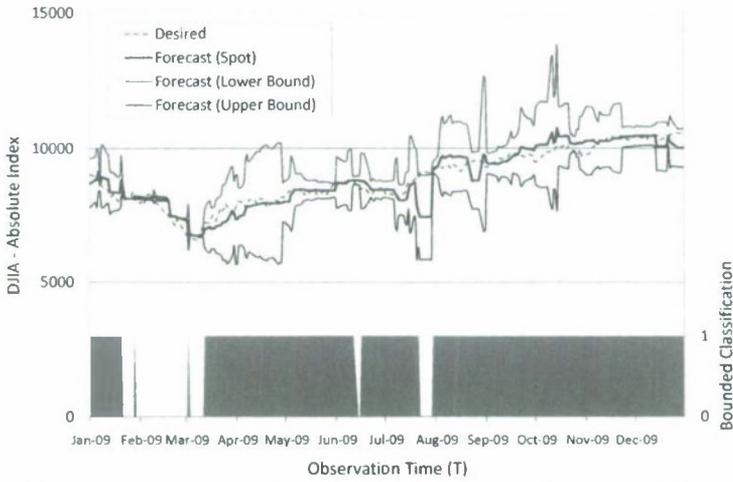
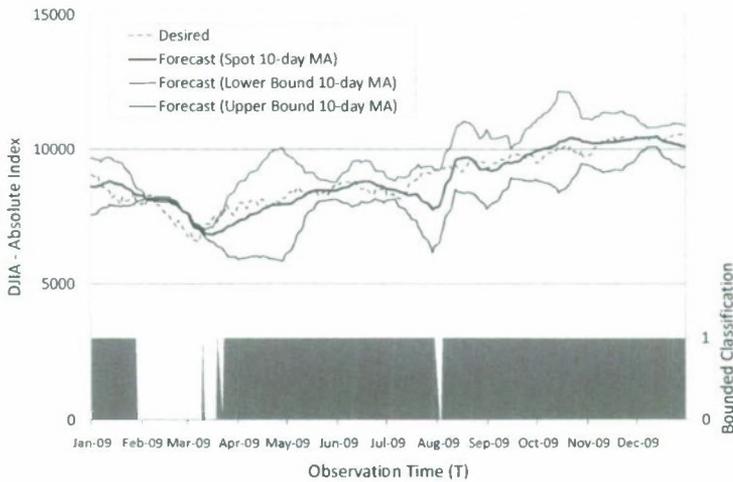


Fig. 7. Self-reorganization of $m(T + 1)$ cluster neurons over 30 years



(a) Interval-forecasting (without moving average) for Year 2009.



(b) Interval-forecasting (with moving average) for Year 2009.

Fig. 8. Interval-forecasting with/without moving average smoothing

associations on movements in the DJIA about once every two years, which is rather consistent with the experimental observations in [22]. Also, while the rule count appears to fluctuate a lot in Fig. 5, the visualization for the year 2009 in Fig. 6(b) shows that the online rule-learning process is rather gradual.

The BCM rule learning process is supported by a self-reorganizing of cluster algorithm described [22]. Reorganization of the cluster nodes occurs in three aspects: (a) cluster widths, (b) cluster centroids and (c) number of clusters nodes. Fig. 7 provides a visual summary of how the clusters shift and spread into new data regions over three decades. In addition, the bottom half of Fig. 7 shows that about twenty output $m(T + 1)$ clusters are used on average.

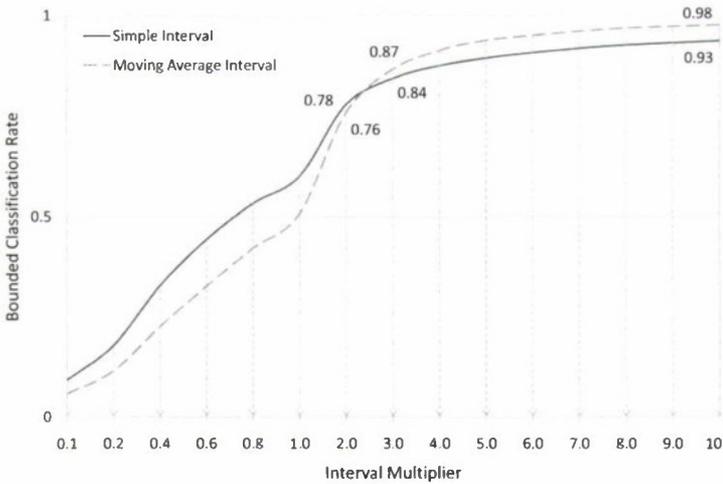


Fig. 9. Interval-multiplier effect on bounded classification rate

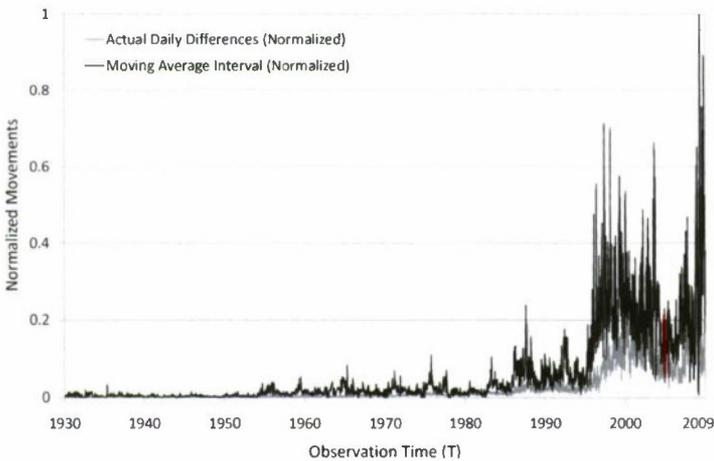


Fig. 10. Correlation between normalized moving average interval and daily differences

Next, the interval-forecasting based on eqn. (5) for $k = 3$ is examined for the DJIA index. From iSeroFAM, the interval-forecasts for Year 2009 are generated with the lower bounds, upper bounds and spot forecasts as shown in Fig. 8(a). At any point in time, when the “desired” output falls within the lower and upper bounds of forecasts, the output is considered to be classified correctly. When the “desired” output falls out of the bounds interval-forecast, the output is considered to be classified incorrectly. The bottom half of Fig. 8(a) indicates the bounded classification as ‘1’ when the output is correctly classified, and ‘0’ when the output is incorrectly classified. In this case, the bounded classification is about 84% for the eighty years of DJIA interval-forecasting.

To smoothen the interval-forecasting from Fig. 8(a), a ten-day moving average of the interval-forecasts were computed that is shown in Fig. 8(b). As can be seen, the interval-forecasting with moving average improves the readability of the lower and upper bounds of the interval-forecasts. With the moving average, the bounded classification rises to about 87% of outputs falling within the bounded classification.

Logically, the bounded classification rate is affected by the interval-multiplier specified in eqn. (5). The larger the multiplier, the higher the bounded classification rate. Fig. 9 examines the impact of the interval-multiplier with respect to the bounded classification for the interval-forecasts, with and without moving average. As can be seen, the moving average interval-forecasts work only better at higher multiplier values. On the other hand, as mentioned earlier for Fig. 8(b), the moving average approach provides improved visualization of the interval-forecasting. In addition, it has been noted that there is a positive correlation of 0.71 between the actual daily differences $d(T + 1) = m(T + 1) - m(T)$, and the moving average interval-forecasts on a normalized basis. This is interesting for further detailed study because it presents, on a preliminary basis, that the moving average interval-forecasts could provide some forecast of the real volatility risk of the DJIA index.

4 Summary and Conclusion

This paper presents iSeroFAM, an online self-reorganizing neuro-fuzzy approach that is based on the BCM theory of metaplasticity. BCM theory accounts for temporal shifts in learning online patterns through a self-correcting associative and dissociative learning mechanism. The experimental proof-of-concept for the iSeroFAM approach was based on the real-world DJIA Index. Preliminary findings show that iSeroFAM reorganizes its rules and clusters about once in two years to meet changing environmental conditions. Also, moving average based interval-forecasting appear to be a useful variability indicator of real volatility in the DJIA market index.

References

1. Abraham, W.C., Goddard, G.V.: Asymmetric relationships between homosynaptic long-term potentiation and heterosynaptic depression. *Nature* 305, 717–719 (1983)
2. Angelov, P.P., Filev, D.P.: An approach to online identification of Takagi-Sugeno fuzzy models. *IEEE Trans. Syst. Man Cybern. B* 34(1), 484–498 (2004)
3. Angelov, P.P., Filev, D.P.: SimLeTS: a simplified method for learning evolving Takagi-Sugeno fuzzy models. In: *Proc. 14th IEEE Int. Conf. Fuzzy Syst., FUZZ-IEEE 2005*, Reno, NV, pp. 1068–1073 (2005)
4. Bienenstock, E.L., Cooper, L.N., Mumro, P.W.: A theory for the development of neuron selectivity: orientation specificity and binocular interaction in visual cortex. *J. Neurosci.* 2, 32–48 (1982)
5. Bliss, T.V.P., Lomo, T.: Long-lasting potentiation of synaptic transmission in the dentate area of the anaesthetized rabbit following stimulation of the perforant path. *J. Physiol.* 232, 331–356 (1973)

6. Carlsson, C., Fullér, R.: On possibilistic mean value and variance of fuzzy numbers. *Fuzzy Sets Syst.* 122(2), 315–326 (2001)
7. Cooper, L.N., Intrator, N., Blais, B.S., Shouval, H.Z.: *Theory of Cortical Plasticity*. World Scientific, Singapore (2004)
8. Goodwin, G.C., Sin, K.S.: *Adaptive Filtering: Prediction and Control*. Prentice-Hall, Englewood Cliffs (1984)
9. Haykin, S.: *Adaptive Filter Theory*. Prentice-Hall, Englewood Cliffs (1996)
10. Jang, J.S.R.: ANFIS: adaptive-network-based fuzzy inference system. *IEEE Trans. Syst. Man Cybern.* 23(3), 665–685 (1993)
11. Jang, J.S.R., Sun, C.T., Mizutani, E.: *Neuro-Fuzzy and Soft Computing*. Prentice Hall, Upper Saddle River (1997)
12. Kasabov, N.K.: Evolving fuzzy neural networks for supervised/unsupervised online knowledge-based learning. *IEEE Trans. Syst. Man Cybern. B* 31(6), 902–918 (2001)
13. Kasabov, N.K., Song, Q.: DENFIS: dynamic evolving neural-fuzzy inference system and its application for time-series prediction. *IEEE Trans. Fuzzy Syst.* 10(2), 144–154 (2002)
14. Lalla, M., Facchinetti, G., Mastroleo, G.: Vagueness evaluation of the crisp output in a fuzzy inference system. *Fuzzy Sets Syst.* (2008) (in Press)
15. Leng, G., Prasad, G., McGinnity, T.M.: An on-line algorithm for creating self-organizing fuzzy neural networks. *Neural Netw.* 17, 1477–1493 (2004)
16. Lin, C.T., Lee, C.S.G.: *Neural Fuzzy Systems: A Neuro-Fuzzy Synergism to Intelligent Systems*. Prentice Halls, Upper Saddle River (1996)
17. Mitra, S., Hayashi, Y.: Neuro-fuzzy rule generation: survey in soft computing framework. *IEEE Trans. Neural Netw.* 11(3), 748–768 (2000)
18. Rochester, N., Holland, J., Haibt, L., Duda, W.: Tests on a cell assembly theory of the action of the brain, using a large scale digital computer. *IRE Trans. Inf. Theory* IT-2, 80–93 (1956)
19. Rong, H.J., Sundararajan, N., Huang, G.B., Saratchandran, P.: Sequential adaptive fuzzy inference system (SAFIS) for nonlinear system identification and prediction. *Fuzzy Sets Syst.* 157, 1260–1275 (2006)
20. Stanton, P.K., Sejnowski, T.J.: Associative long-term depression in the hippocampus induced by Hebbian covariance. *Nature* 339, 215–218 (1989)
21. Tan, J., Quek, C.: Adaptive training schema in Mamdani-type neuro-fuzzy models for data-analysis in dynamic system forecasting. In: Liu, D. (ed.) *Proc. 2008 Int. Joint Conf. Neural Netw., IJCNN 2008, HongKong*, pp. 1734–1739 (2008)
22. Tan, J., Quek, C.: A BCM-theory of meta-plasticity for online self-reorganizing fuzzy-associative learning. *IEEE Trans. Neural Netw.* 21(6), 985–1003 (2010)
23. Tickle, A.B., Andrews, R., Golea, M., Diederich, J.: The truth will come to light: directions and challenges in extracting the knowledge embedded within trained artificial neural networks. *IEEE Trans. Neural Netw.* 9(6), 1057–1068 (1998)
24. Wang, W., Vrbanek, J.: A multi-step predictor for dynamic system property forecasting. *Measurement Sci. Technol.* 18(12), 3673–3681 (2007)
25. Watts, M.J.: A decade of Kasabov's evolving connectionist systems: a review. *IEEE Trans. Syst. Man Cybern. C, Appl. Rev.* 39(3), 253–269 (2009)
26. Wu, S., Er, M.J.: Dynamic fuzzy neural networks: a novel approach to function approximation. *IEEE Trans. Syst. Man Cybern. B* 30(2), 358–364 (2000)
27. Zadeh, L.A.: Soft computing and fuzzy logic. *IEEE Softw.* 11(6), 48–56 (1994)

An Evolving Type-2 Neural Fuzzy Inference System

Sau Wai Tung¹, Chai Quek^{1,*}, and Cuntai Guan²

¹ Center for Computational Intelligence, Sch. of Comp. Engineering,
Nanyang Technological University, Singapore

ashcquek@ntu.edu.sg

² Institute for Infocomm Research, A*Star, Singapore

Abstract. Traditional designs of neural fuzzy systems are largely user-dependent whereby the knowledge to form the computational structures of the systems is provided by the user. By designing a neural fuzzy system based on experts' knowledge results in a non-varying structure of the system. To overcome the drawback of a heavily user-dependent system, self-organizing methods that are able to directly utilize knowledge from the numerical training data have been incorporated into the neural fuzzy systems to design the systems. Nevertheless, this data-driven approach is insufficient in meeting the challenges of real-life application problems with time-varying dynamics. Hence, this paper is a novel attempt in addressing the issues involved in the design for an evolving Type-2 Mamdani-type neural fuzzy system by proposing the *evolving Type-2 neural fuzzy inference system* (eT2FIS) – an online system that is able to fulfill the requirements of evolving structures and updating parameters to model the non-stationeries in real-life applications.

Keywords: Evolving systems, online systems, neural fuzzy systems, incremental sequential learning, Type-2 fuzzy systems.

1 Introduction

There are two main issues to consider in the design of a neural fuzzy system: (1) the fuzzy partitionings of the input-output dimensions and (2) the generation of the fuzzy rulebase of the system. Traditionally, the design of a neural fuzzy system is largely user-dependent whereby both the fuzzy partitioning and the rulebase of the system are manually crafted by human experts. The structure of the neural fuzzy system is fixed once the necessary knowledge has been determined by the experts, and only the parameters of the system are updated in subsequent training. In order to minimize the dependency on subjective information from human users, numerical methods such as fuzzy Kohonen partitioning [2], fuzzy C-means [1] and linear vector quantization [9] were incorporated into the systems to directly acquire knowledge from the numerical training data to perform fuzzy partitioning. In addition, self-organizing rule generation

* Corresponding author.

schemes [13] [15] [12] were also proposed to overcome the knowledge acquisition bottleneck. This subsequently leads to a new class of neural fuzzy systems with self-organizing abilities that are able to directly utilize knowledge from the numerical training data to design the computational structures of the systems.

Nevertheless, the demands and complexities of real-life applications often require the neural fuzzy system to be able to adapt not just its parameters, but also its structure in order to model the changing dynamics of the application environments. Subsequently, this leads to an intense research effort in the studies of evolving/online neural fuzzy systems which are able to adapt both the structures and the parameters of the systems to model such time-varying dynamics. Depending on the formulation of the set of fuzzy rules that governs the computational structure of the network, there are generally two classes of evolving neural fuzzy systems, mainly the Takagi-Sugeno-Kang (TSK) systems [3] [8] [4] [5] and the Mamdani systems [10] [7] [14]. Most of the existing work in the literature consists of evolving Type-1 TSK-type and Type-1 Mamdani-type neural fuzzy systems. These models may not perform adequately under noisy application environments when compared to their Type-2 counterparts due to the use of crisp membership grades. Hence, there have been recent efforts to extend the evolving Type-1 TSK-type neural fuzzy systems to Type-2 systems as seen by the emergence of the SEIT2FNN [4] and the ORGQACO [5] models. In contrast, there has been no such attempt in the parallel track for evolving Type-1 Mamdani-type neural fuzzy systems.

This paper is a novel attempt in synergizing the individual frameworks of evolving systems and Type-2 Mamdani-type neural fuzzy systems by presenting the *evolving Type-2 neural fuzzy inference system* (eT2FIS). The proposed eT2FIS model adopts a two phase incremental sequential learning scheme whereby the neural fuzzy system performs structural learning and parameter learning upon the arrival of each new training data point. Initially, there are no fuzzy partitioning or fuzzy rules in the system, i.e., there are no hidden nodes in the network. Subsequently, the computational structure of the neural fuzzy system, which is governed by a set of Type-2 IF-THEN Mamdani rules, is incrementally formulated based on the knowledge from each training data point. There are three key operations contained in the structural learning phase of the system: (1) the generation of new fuzzy rules, (2) the deletion of obsolete rules, and (3) the merger of highly similar fuzzy labels; while parameter learning is performed using the neural-network based backpropagation mechanism.

The rest of the paper is organized as follows. The structure and the operations of eT2FIS are described in Section 2. Section 3 presents the online learning mechanism of eT2FIS. The adaptation abilities of the system are evaluated in Section 4. Section 5 concludes the paper.

2 eT2FIS: Architecture and Neural Operations

The eT2FIS is a five layers neural fuzzy system as shown in Fig. 1. Layer 1 of the system consists of the input linguistic nodes; layer 2 consists of the antecedent nodes; layer 3 is the rule nodes; layer 4 is the consequent nodes; and

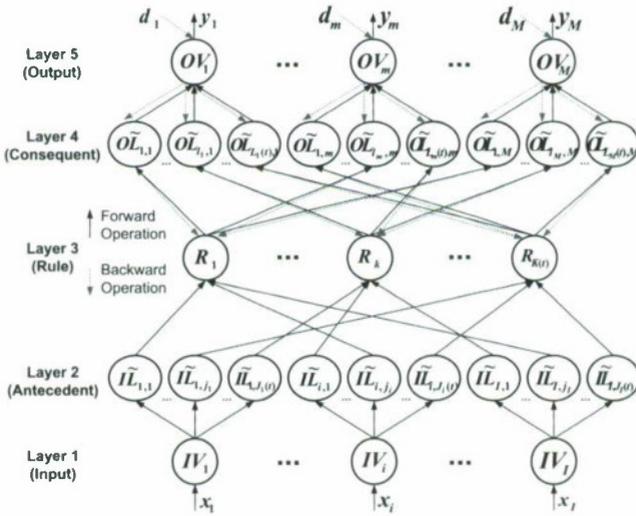


Fig. 1. Architecture of the evolving Type-2 neural fuzzy inference system (eT2FIS)

layer 5 consists of the output linguistic nodes. As mentioned before, the initial form of the neural fuzzy system consists of no hidden layers and learning for the system is performed incrementally when each training tuple $[X(t); D(t)]$ is presented to the system one at a time where $X(t) = (x_1(t), \dots, x_i(t), \dots, x_I(t))$ and $D(t) = (d_1(t), \dots, d_m(t), \dots, d_M(t))$ represent the vectors for the input training data and the corresponding desired output data at time step t respectively. Each input node $IV_i, i \in \{1 \dots I\}$, in layer 1 of the system takes in a single input value x_i from the input training vector and subsequently, each output node $OV_m, m \in \{1 \dots M\}$, in layer 5 produces a single output value y_m where the corresponding computed output vector is represented as $Y(t) = (y_1(t), \dots, y_m(t), \dots, y_M(t))$. During time step t , each input node will consist of $J_i(t)$ number of corresponding fuzzy labels in layer 2 of the system such that each antecedent node is represented as $\tilde{I}_{i,j_i}, j_i \in \{1 \dots J_i(t)\}$. Similarly, each output node will consist of $L_m(t)$ number of corresponding fuzzy labels in layer 4 of the system such that each consequent node is represented as $\tilde{O}_{l_m,m}, l_m \in \{1 \dots L_m(t)\}$. The connectionist structure of the proposed model is based on a set of fuzzy rules $R_k, k \in \{1 \dots K(t)\}$, defined in layer 3 of the system. In the proposed model, the number of rules $K(t)$, the number of fuzzy labels for the i -th input variable $J_i(t)$, and the number of fuzzy labels for the m -th output variable $L_m(t)$ vary with the changes in the underlying dynamics of the application environment.

For the proposed eT2FIS model, the training parameters are the centers of the left and right formation gaussian functions of the interval Type-2 fuzzy labels present in layers 2 and 4 of the network as shown in Fig. 2. Each fuzzy label in the antecedent layer and consequent layer is defined by its footprint of uncertainty [11] $\mu_{\tilde{A}}(x) = [\underline{\mu}_{\tilde{A}}(x), \bar{\mu}_{\tilde{A}}(x)]$ where \tilde{A} denotes the Type-2 fuzzy set.

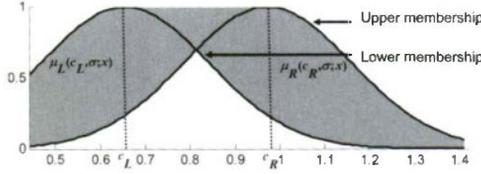


Fig. 2. An interval Type-2 fuzzy set in the antecedent/consequent layer denoted by its left and right formation gaussian functions

Subsequently, the lower and upper membership functions of \tilde{A} are defined as

$$\underline{\mu}_{\tilde{A}}(x) = \begin{cases} \mu_R(c_R, \sigma; x) & \text{if } x \leq \frac{c_L + c_R}{2} \\ \mu_L(c_L, \sigma; x) & \text{otherwise} \end{cases} \quad \text{and} \quad \overline{\mu}_{\tilde{A}}(x) = \begin{cases} \mu_L(c_L, \sigma; x) & \text{if } x \leq c_L \\ 1 & \text{if } c_L < x \leq c_R \\ \mu_R(c_R, \sigma; x) & \text{if } x > c_R \end{cases}$$

respectively. Here, $\mu_L(c_L, \sigma; x)$ and $\mu_R(c_R, \sigma; x)$ refer to the left and right formation gaussian functions respectively such that they are defined based on the underlying gaussian function $\mu(c, \sigma; x) = e^{-((x-c)^2/\sigma^2)}$ where c is the centre of the function and σ is the width of the function.

Next, the two key neural operations in the proposed model, namely the forward and the backward computations, are described as follows.

2.1 Forward Operations

The forward aggregated input and output for an arbitrary node are denoted as NET and Z respectively.

Layer 1: $NET_{IV_i} = Z_{IV_i} = x_i$.

Layer 2: $NET_{\tilde{L}_{i,j_i}} = x_i$ and $Z_{\tilde{L}_{i,j_i}} = [\underline{f}_{i,j_i}, \overline{f}_{i,j_i}]$ such that the similarity between the input value x_i and the respective fuzzy labels is an interval Type-1 set with bounds defined by $\underline{f}_{i,j_i} = \begin{cases} \mu_{R_{i,j_i}}(c_{R_{i,j_i}}, \sigma; x_i) & \text{if } x_i \leq \frac{c_{L_{i,j_i}} + c_{R_{i,j_i}}}{2} \\ \mu_{L_{i,j_i}}(c_{L_{i,j_i}}, \sigma; x_i) & \text{otherwise} \end{cases}$ and

$$\overline{f}_{i,j_i} = \begin{cases} \mu_{L_{i,j_i}}(c_{L_{i,j_i}}, \sigma; x_i) & \text{if } x_i \leq c_{L_{i,j_i}} \\ 1 & \text{if } c_{L_{i,j_i}} < x_i \leq c_{R_{i,j_i}} \\ \mu_{R_{i,j_i}}(c_{R_{i,j_i}}, \sigma; x_i) & \text{if } x_i > c_{R_{i,j_i}} \end{cases} \text{ respectively.}$$

Layer 3: $NET_{R_k} = \left\{ \left[\underline{f}_{i,j_i}^{(k)}, \overline{f}_{i,j_i}^{(k)} \right] \right\}$ and $Z_{R_k} = [\underline{f}_k, \overline{f}_k]$ where the overall similarity between the input vector and the antecedent segment of the k -th fuzzy rule is an interval Type-1 set with bounds given as $\underline{f}_k = \min_{i \in \{1 \dots J\}} \underline{f}_{i,j_i}^{(k)}$ and $\overline{f}_k = \min_{i \in \{1 \dots J\}} \overline{f}_{i,j_i}^{(k)}$ respectively.

Layer 4: $NET_{\tilde{O}_{L_{l_m,m}}} = \left\{ \left[\underline{f}_k, \overline{f}_k \right] \right\}$ and $Z_{\tilde{O}_{L_{l_m,m}}} = [\underline{f}_{l_m,m}, \overline{f}_{l_m,m}]$ where $\underline{f}_{l_m,m} = \max_{k \in K_{l_m,m}} \underline{f}_k$ and $\overline{f}_{l_m,m} = \max_{k \in K_{l_m,m}} \overline{f}_k$ respectively. Here, $K_{l_m,m}$ is the set of fuzzy rules in the system that share the same output label $\tilde{O}_{L_{l_m,m}}$.

Layer 5: $NET_{OV_m} = Y_m$ and $Z_{OV_m} = y_m$ where the type-reduced set obtained using the height-type-reduction (HTR) [6] is an interval Type-1 set $Y_m := \int_{\rho_{1,m}} \dots \int_{\rho_{L_m,m}} 1 / \frac{\sum_{l_m=1}^{L_m(t)} y_{l_m,m}^* \rho_{l_m,m}}{\sum_{l_m=1}^{L_m(t)} \rho_{l_m,m}} = [Y_m^{\min}, Y_m^{\max}]$. Here, $y_{l_m,m}^*$ is defined to be the midpoint of the domain of $\tilde{O}L_{l_m,m}$, and $\rho \in [\underline{f}_{l_m,m}, \bar{f}_{l_m,m}]$. Then the computed output is given to be the defuzzified value $y_m = \frac{1}{2} [Y_m^{\min} + Y_m^{\max}]$.

2.2 Backward Operations

The backward operation of the eT2FIS model, as represented by the dotted arrows in Fig. 1, from layer 5 to layer 3 of the system is a mirrored computation of the forward operation. Correspondingly, the backward aggregated input and output for an arbitrary node are denoted as NET^{back} and Z^{back} respectively.

Layer 5: $NET_{OV_m}^{\text{back}} = Z_{OV_m}^{\text{back}} = d_m$.

Layer 4: $NET_{\tilde{O}L_{l_m,m}}^{\text{back}} = d_m$ and $Z_{\tilde{O}L_{l_m,m}}^{\text{back}} = [\underline{f}_{l_m,m}^{\text{back}}, \bar{f}_{l_m,m}^{\text{back}}]$ such that the

bounds are defined as $\underline{f}_{l_m,m}^{\text{back}} = \begin{cases} \mu_{R_{l_m,m}}(c_{R_{l_m,m}}, \sigma; d_m) & \text{if } d_m \leq \frac{c_{L_{l_m,m}} + c_{R_{l_m,m}}}{2} \\ \mu_{L_{l_m,m}}(c_{L_{l_m,m}}, \sigma; d_m) & \text{otherwise} \end{cases}$

and $\bar{f}_{l_m,m}^{\text{back}} = \begin{cases} \mu_{L_{l_m,m}}(c_{L_{l_m,m}}, \sigma; d_m) & \text{if } d_m \leq c_{L_{l_m,m}} \\ 1 & \text{if } c_{L_{l_m,m}} < d_m \leq c_{R_{l_m,m}} \\ \mu_{R_{l_m,m}}(c_{R_{l_m,m}}, \sigma; d_m) & \text{if } d_m > c_{R_{l_m,m}} \end{cases}$ respectively.

Layer 3: $NET_{R_k}^{\text{back}} = \left\{ \left[(\underline{f}_{l_m,m}^{\text{back}})^{(k)}, (\bar{f}_{l_m,m}^{\text{back}})^{(k)} \right] \right\}$ and $Z_{R_k}^{\text{back}} = [\underline{f}_k^{\text{back}}, \bar{f}_k^{\text{back}}]$

where $\underline{f}_k^{\text{back}} = \min_{m \in \{1 \dots M\}} (\underline{f}_{l_m,m}^{\text{back}})^{(k)}$ and $\bar{f}_k^{\text{back}} = \min_{m \in \{1 \dots M\}} (\bar{f}_{l_m,m}^{\text{back}})^{(k)}$ respectively.

The backward neural computation of the eT2FIS is defined to (1) calculate the *certainty factors* of the fuzzy rules, and (2) determine the creation of a new fuzzy rule (refer to Section 3) when each training tuple $[X(t); D(t)]$ is presented to the system. The certainty factor of a fuzzy rule in the system, as defined in (1), reflects the potential of the rule in describing the current underlying dynamics of the application environment.

$$Cer_k(t) := \max[\text{Age}_k(t), \text{Act}_k(t)] ; Cer_k(0) := 1 \tag{1}$$

where $\text{Age}_k(t) := \eta_k \cdot Cer_k(t - 1)$ constitutes the forgetting component and $\text{Act}_k(t) := \min \left[\frac{1}{2} [\underline{f}_k + \bar{f}_k], \frac{1}{2} [\underline{f}_k^{\text{back}} + \bar{f}_k^{\text{back}}] \right]$ constitutes the enhancement component to the certainty factor. Initially, the certainty factor for a newly formed fuzzy rule is set as unity. This means that the newly created rule is assigned the highest degree of faith in its ability to model the application environment since $0 < Cer_k \leq 1$. As time progresses, the determination of the

certainty factor of a rule is either dominated by the forgetting component Age_k or the enhancement component Act_k . If a rule in the system is able to generalize the recent encountered set of training data well, the dominating factor in the calculation of its certainty factor is the enhancement component. This subsequently enables the computed certainty factor to be of a high value, thus ensuring that the rule will remain in the fuzzy rulebase of the model. On the other hand, if a rule fails to give a satisfactory representation of the current set of encountered training data, then the forgetting mechanism kicks in. Subsequently, the faith in the rule decreases gradually over time until it becomes invalid to the application or it gets recovered through a rehearsal episode. Hence, through this incremental update of the certainty factors for the fuzzy rules in the proposed model, the system is ensured a current and up-to-date set of rulebase that is able to model the underlying dynamics of the application environment.

3 Incremental Learning in eT2FIS

The proposed eT2FIS model adopts a two phase incremental learning process, namely the structural learning and the parameter learning, as shown in Fig. 3. Three key operations are contained within the structural learning phase of the system: (1) the generation of new fuzzy rules, (2) the deletion of obsolete rules, and (3) the merger of highly over-lapping/similar fuzzy labels; while parameter learning is performed using the neural-network based backpropagation mechanism. The initial neural fuzzy system is empty, i.e. there are no hidden layers, and learning is performed incrementally where each training tuple is presented to the system individually at time step t . When the first training sample $[X(0); D(0)]$ arrives, the knowledge from the training data point is used to initialize the system by forming the fuzzy labels $\tilde{I}L_{i,1}$ and $\tilde{O}L_{1,m}$ such that the centres of the left and right functions of the new fuzzy labels are set to be the corresponding input and output values from the vectors $X(0)$ and $D(0)$. The width σ is fixed in the functions. In addition to establishing the fuzzy partitionings in the input-output dimensions of the system, a new fuzzy rule is also created to encode the knowledge represented by the training data point where the antecedent and consequent segments of the new fuzzy rule are defined by the respective sets of newly created fuzzy labels $\left\{ \tilde{I}L_{i,1} \right\}_{i=1}^I$ and $\left\{ \tilde{O}L_{1,m} \right\}_{m=1}^M$. On the other hand, the online structural and parameter learning process of the system are activated by an incoming training tuple if there are existing rules in the system, and the system evolves and learns based on the information provided by the new training tuple. This section describes the online learning mechanism of the eT2FIS.

3.1 Structural Learning

The three main operations in the structural learning phase of the system are described as follows.

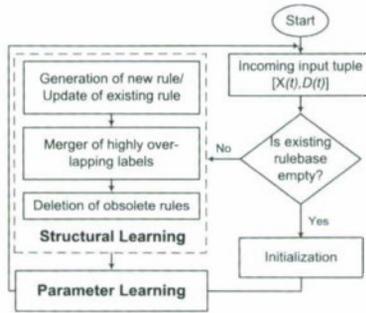


Fig. 3. Flowchart of the incremental learning process in eT2FIS

Creation of new rule: The structural learning phase of the proposed model is activated with the arrival of a new training data sample $[X(t); D(t)]$ to a non-empty neural fuzzy system. If there exists a fuzzy rule R_{k^*} in the current rulebase of the system such that it is able to represent the training sample competently, the system proceeds on to the next stage of the structural learning phase. This is determined by the condition $R_{k^*}(X(t), D(t)) > RuleGen$, $k^* = \arg \max_{k \in \{1 \dots K(t)\}} R_k(X(t), D(t))$, where the activation of the rule R_k by $[X(t); D(t)]$ is given as $R_k(X(t), D(t)) := \min \left[\frac{1}{2} [\underline{f}_k + \bar{f}_k], \frac{1}{2} [\underline{f}_k^{back} + \bar{f}_k^{back}] \right]$ and $RuleGen$ is a pre-defined rule creation threshold. In this paper, $RuleGen$ is fixed as a constant 0.6. Subsequently, the certainty factors for the fuzzy rules are updated using (1) and the system moves on to the second operation in the structural learning phase.

On the other hand, if none of the rules in the system is able to give a satisfactory representation of the training data point, a new fuzzy rule R' is created to encrypt the knowledge from the training sample. The system proceeds by finding the best matched fuzzy labels \tilde{I}_{i,j_i^*} and $\tilde{O}_{L_m^*,m}$ to the data point $[X(t); D(t)]$

where
$$\begin{cases} j_i^* = \arg \max_{j_i \in \{1 \dots J_i(t)\}} \frac{1}{2} [\underline{f}_{i,j_i} + \bar{f}_{i,j_i}] \\ l_m^* = \arg \max_{l_m \in \{1 \dots L_m(t)\}} \frac{1}{2} [\underline{f}_{l_m,m}^{back} + \bar{f}_{l_m,m}^{back}] \end{cases}$$
 . Subsequently, each of

the labels in the set of best matched fuzzy labels can be categorised into three operations as follows:

1. No action is required for the best matched fuzzy label and it is defined as part of the antecedent/consequent segment of the rule R' . This scenerio occurs when the match between the input value $x_i(t)$ (resp. output value $d_m(t)$) and the corresponding best matched label \tilde{I}_{i,j_i^*} (resp. $\tilde{O}_{L_m^*,m}$) is highly similar, i.e., $\frac{1}{2} [\underline{f}_{i,j_i^*} + \bar{f}_{i,j_i^*}] > 0.75$ or $\frac{1}{2} [\underline{f}_{l_m^*,m}^{back} + \bar{f}_{l_m^*,m}^{back}] > 0.75$.
2. No action is required for the best matched label and a new label is created as part of the antecedent/consequent segment of the rule R' . This scenerio occurs when the similarity between the input-output value and its corresponding best

matched label is minimal, i.e., $\frac{1}{2} [f_{i,j_i^*} + \bar{f}_{i,j_i^*}] < 0.25$ or $\frac{1}{2} [f_{l_m^*,m}^{\text{back}} + \bar{f}_{l_m^*,m}^{\text{back}}] < 0.25$. A new fuzzy label $\tilde{L}_{i,J_i(t+1)}$, $J_i(t+1) = J_i(t) + 1$, or $\tilde{O}L_{L_m(t+1),m}$, $L_m(t+1) = L_m(t) + 1$, is created such that the centres of the left and right functions of the new fuzzy label is set to be the corresponding input-output value from the training vector.

3. *The spread of the best matched fuzzy label is expanded and the expanded label is defined as part of the antecedent/consequent segment of the rule R'.* This scenario occurs when the similarity between the input-output value and its corresponding best matched label falls in the interval $[0.25, 0.75]$, i.e., the match is reasonable but not satisfactory. Subsequently, the best matched label will expand itself by increasing the spread s between the centres of the left and right functions of the fuzzy label to incorporate the current input-output value by $\begin{cases} s_{i,j_i^*}(t+1) = \min [s_{\max}, s_{i,j_i^*}(t) + \eta_s \cdot s_{\max}] \\ s_{l_m^*,m}(t+1) = \min [s_{\max}, s_{l_m^*,m}(t) + \eta_s \cdot s_{\max}] \end{cases}$ such that s_{\max} is the maximum permissible spread for each of the fuzzy label.

Although a new fuzzy rule R' has been created as described above, it will only be included in the rulebase of the neural fuzzy system if it is not ambiguous and the novelty of the fuzzy rule is ascertained.

Merger of Highly Over-Lapping Fuzzy Labels: The second stage in the structural learning phase of the proposed model is the merging of two highly similar/over-lapping fuzzy labels in each of the input-output dimensions. If the similarity measure between two interval Type-2 fuzzy labels \tilde{A}_1 and \tilde{A}_2 , $SM(\tilde{A}_1, \tilde{A}_2)$, is greater than a merger threshold δ_{MF} , the fuzzy labels \tilde{A}_1 and

\tilde{A}_2 are merged such that $\begin{cases} c_{L_1} = \frac{c_{L_1} + c_{L_2}}{2}, c_{R_1} = \frac{c_{R_1} + c_{R_2}}{2} \\ \{\tilde{A}_p\}(t+1) = \{\tilde{A}_p\}(t) \setminus \tilde{A}_2 \end{cases}$ where $\{\tilde{A}_p\}(t)$ is the set of fuzzy labels in the corresponding input-output dimension at time step t , and c_{L_1} and c_{R_1} are the centres of the left and right functions of the Type-2 label \tilde{A}_1 respectively.

Deletion of Obsolete Rule: The final stage in the structural learning phase of the proposed model is the deletion of any obsolete rules that are present in the rulebase of the system at time step t . A fuzzy rule in the neural fuzzy system is regarded as an invalid/out-dated rule in the system if the certainty factor of the rule falls below a threshold $RuleDel$ where $RuleDel$ represents the minimum potential that a fuzzy rule should possess for it to be considered having the ability to model the current underlying dynamics of the application environment. In this paper, $RuleDel$ is fixed as a constant 0.35. Subsequently, $K(t+1) = K(t) - 1$.

The combination of the three operations within the framework of the proposed eT2FIS model ensures that the neural fuzzy system maintains a set of up-to-date and compact fuzzy rulebase that is able to model the current underlying dynamics of the application. This is because a new rule is created when the new training data point cannot be represented satisfactorily by the existing

set of fuzzy rules in the system; and obsolete rules are deleted when they are no longer valid under the current application environment. In addition, highly over-lapping/similar fuzzy labels in each of the input-output dimensions are also merged to reduce the computational complexity of the neural fuzzy system and this helps to improve the overall interpretability of the system.

3.2 Parameter Learning

The second phase in the incremental sequential learning process of the proposed model is parameter learning. After a newly arrived training data sample $[X(t); D(t)]$ passes through the structural learning phase, it will activate parameter learning in the system where the objective of the parameter adaptation is to minimize the difference in error between the computed output $Y(t)$ and the desired output $D(t)$ at each time step t . The error function at time t is thus defined as $E = \frac{1}{2} \sum_{m=1}^M [d_m - y_m]^2$. Parameter adaptation in the proposed eT2FIS is performed based on a neural-network based backpropagation mechanism.

4 Experimental Results

This section describes two experimental simulations performed by eT2FIS, namely system identification of a time-varying plant and that of a plant with noise.

4.1 System Identification of a Time-Varying Plant

To illustrate the abilities to evolve and adapt, the eT2FIS model is employed to model the underlying characteristics of a time-varying plant as described in [4]: $y(t+1) = \frac{y(t)}{1+y^2(t)} + u^3(t) + f(t)$, $u(t) = \sin(2\pi t/100)$ where $f(t) = \begin{cases} 0, & t \leq 1000 \text{ and } t \geq 2001 \\ 1, & 1001 \leq t \leq 2000 \end{cases}$. The initial conditions $(u(0), y(0))$ are set as $(0, 0)$ and the objective of the experiment is to identify the output $y(t+1)$ given the input vector $(u(t), y(t))$ at each time step t . For the purpose of this experiment, 3000 data tuples are produced. The proposed system is employed to identify the plant in an online sequential mode, i.e., there is no prior knowledge of the plant such that the training tuples are presented to the system individually at each time step through a single pass.

Fig. 4(a) illustrates the performance of the eT2FIS model in the modeling of the time-varying plant. Fig. 4(a)(i) shows the number of rules identified by the proposed model during the online identification of the plant. The fluctuations in the number of rules identified at the start of the experiment, the start of $t = 1000$ (when a disturbance $f(t)$ is added) and the start of $t = 2000$ (when $f(t)$ is removed) indicate that the model is trying to learn the underlying characteristics of the plant. After which, the number of identified rules stabilizes before any changes are detected in the underlying dynamics of the environment. Fig. 4(a)(ii) shows the total number of rules identified for the modeling of the time-varying plant achieved by eT2FIS and the benchmarking models, namely

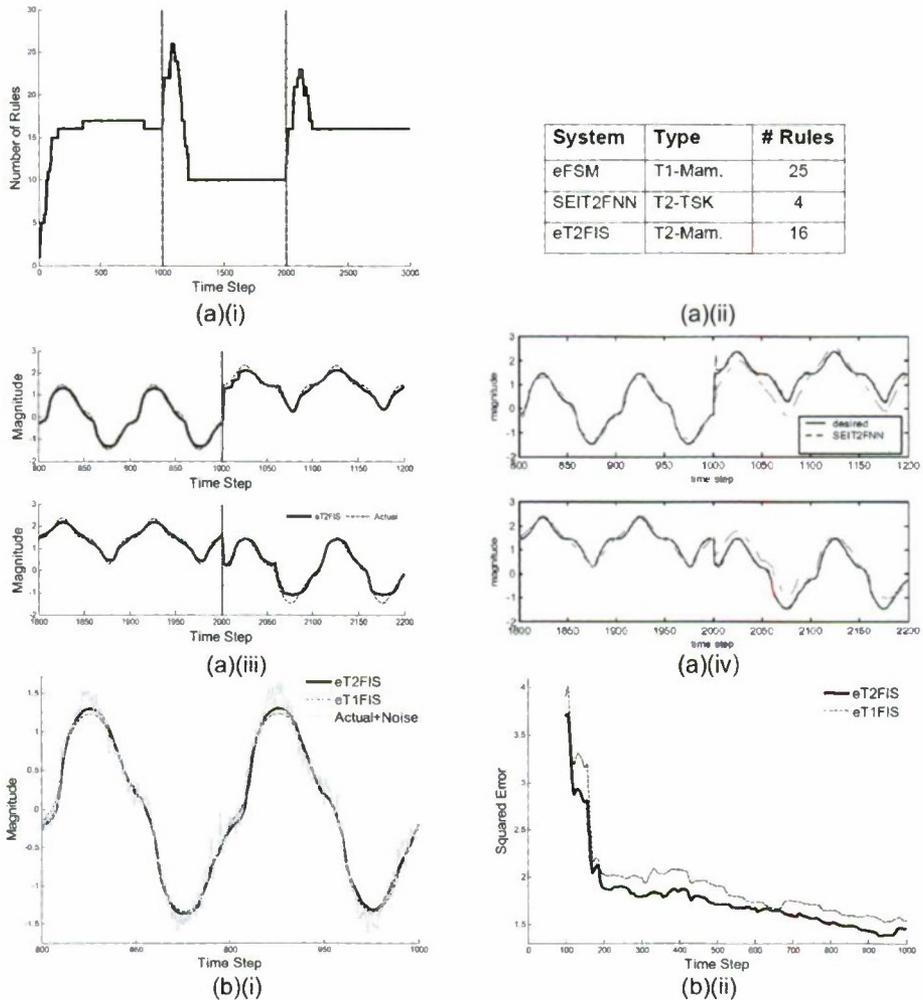


Fig. 4. (a) Experimental results for the time-varying plant with additive disturbance $f(t)$: (i) Number of rules $K(t)$ identified by eT2FIS at each time instance, (ii) Total number of identified rules obtained for eT2FIS and the benchmarking systems, (iii) Online identification results by eT2FIS, and (iv) Online identification results by SEIT2FNN [4]. (b) Illustrative results for the plant with noise: (i) A realization of the identification results for the plant with noise, and (ii) Average online learning errors for the benchmarking systems.

the eFSM [14] and the SEIT2FNN [4] models. By adopting Type-2 sets in the system, the proposed model requires lesser number of rules to model the plant as compared to the Type-1 Mamdani-type eFSM model. This translates to a computationally less complexed eT2FIS system. On the other hand, it is not surprising that the SEIT2FNN model requires much fewer rules compared to

the eT2FIS model because of the greater computational powers of TSK-type systems. Nevertheless, the proposed eT2FIS model is able to achieve a satisfactory modeling performance as the dynamics of the plant changes over time when compared to the SEIT2FNN model as seen from the online identification results in Figs. 4(a)(iii)–(iv).

4.2 System Identification with Noise

To illustrate the noise resistance abilities of the proposed evolving Type-2 system, the eT2FIS is employed to identify the plant $y(t+1) = \frac{y(t)}{1+y^2(t)} + u^3(t)$, $u(t) = \sin(2\pi t/100)$, $t = 0 \dots 1000$. Here, the measured output $y(t+1)$ is assumed to be contaminated by noise. The added noise is an artificially generated Gaussian white noise with variance 0.1. There are 10 Monte Carlo realizations in this experiment. As conducted in the previous experiment, the computational structure of the eT2FIS is incrementally formulated with the arrival of each training tuple.

Fig. 4(b) shows the performance comparison between the eT2FIS model and the Type-1 eT2FIS model (eT1FIS).¹ Fig. 4(b)(i) shows the learning results of the benchmarking systems for one of the 10 realizations. The computed output of both the eT2FIS and the eT1FIS models do not fluctuate as violently as the actual noisy output, indicating that the neural fuzzy systems possess the abilities to model uncertainties in an application environment. Fig. 4(b)(ii) shows the average learning errors over the 10 realizations for the benchmarking systems. Being a Type-2 system, the proposed model is more resistant to the noise present in the underlying dynamics of the environment as seen by the significantly smaller average squared error (calculated as an accumulation over 100 time steps) of the eT2FIS model as compared to the Type-1 model. This means that while neural fuzzy systems are able to incorporate the effects of uncertainties in the structures of the systems, Type-2 systems are able to demonstrate a greater tolerance compared to their Type-1 counterparts under a noisy application environment.

5 Conclusions

This paper presents the eT2FIS model, an evolving Type-2 Mamdani-type neural fuzzy inference system that is able to learn, evolve and adapt with the changes in the environment that it is modeling. Encouraging performances have been achieved when the system is employed to identify a plant with non-stationary dynamics and a plant with noise.

¹ The Type-1 eT2FIS model, denoted as eT1FIS, refers to a modified version of the proposed eT2FIS model where the fuzzy labels in the antecedent/consequent layers of the network are set as Type-1 fuzzy sets. The learning algorithm of the eT1FIS is similar to that of the proposed model. The purpose of benchmarking against an evolving Type-1 neural fuzzy system with similar learning mechanism is to illustrate the greater uncertainty tolerance of a Type-2 system in a noisy environment when compared to its Type-1 counterpart.

References

1. Bezdek, J.C.: *Pattern Recognition with Fuzzy Objective Function Algorithms*. Plenum, New York (1981)
2. Bezdek, J.C., Tsao, E.C.-K., Pal, N.R.: Fuzzy Kohonen clustering networks. In: *IEEE Conference on Fuzzy Systems*, pp. 1035–1043 (1992)
3. Juang, C.F., Lin, C.T.: An on-line self-constructing neural fuzzy inference network and its applications. *IEEE Transactions on Fuzzy Systems* 6(1), 12–32 (1998)
4. Juang, C.F., Tsao, Y.W.: A self-evolving interval Type-2 fuzzy neural network with online structure and parameter learning. *IEEE Transactions on Fuzzy Systems* 16(6), 1411–1424 (2008)
5. Juang, C.F., Hsu, C.H.: Reinforcement interval Type-2 fuzzy controller design by online rule generation and Q-value-aided ant colony optimization. *IEEE Transactions on Systems, Man and Cybernetics B-39(6)*, 1528–1542 (2009)
6. Karnik, N.N., Mendel, J.M., Liang, Q.: Type-2 fuzzy logic systems. *IEEE Transactions on Fuzzy Systems* 7, 643–658 (1999)
7. Kasabov, N.: Evolving fuzzy neural networks for supervised/unsupervised online knowledge-based learning. *IEEE Transactions on Systems, Man and Cybernetics B-31(6)*, 902–918 (2001)
8. Kasabov, N., Song, Q.: DENFIS: Dynamic evolving neural-fuzzy inference system and its application for time-series prediction. *IEEE Transactions on Fuzzy Systems* 10(2), 144–154 (2002)
9. Kohonen, T.: Self-organized formation of topologically correct feature maps. *Biological Cybernetics* 43, 59–69 (1982)
10. Lin, C.J., Lin, C.T.: An ART-based fuzzy adaptive learning control network. *IEEE Transactions on Fuzzy Systems* 5, 477–496 (1997)
11. Mendel, J.M., John, R.I.B.: Type-2 fuzzy sets made simple. *IEEE Transactions on Fuzzy Systems* 10(2), 117–127 (2002)
12. Mitra, S., Hayashi, Y.: Neuro-fuzzy rule generation: survey in soft computing framework. *IEEE Transactions on Neural Networks* 11(3), 748–768 (2000)
13. Quek, C., Zhou, R.W.: The POP learning algorithms: reducing work in identifying fuzzy rules. *Neural Networks* 14, 1431–1445 (2001)
14. Tung, W.L., Quek, C.: eFSM: A novel online neural-fuzzy semantic memory model. *IEEE Transactions on Neural Networks* 21(1), 136–157 (2009)
15. Wang, L.X., Mendel, J.M.: Generating fuzzy rules by learning from examples. *IEEE Transactions on Systems, Man and Cybernetics* 22, 1414–1427 (1992)

Human Augmented Cognition Based on Integration of Visual and Auditory Information

Woong Jae Won¹, Wono Lee¹, Sang-Woo Ban², Minook Kim³, Hyung-Min Park³,
and Minhoo Lee¹

¹ School of Electrical Engineering and Computer Science, Kyungpook National University,
1370 Sankyuk-Dong, Puk-Gu, Taegu 702-701, Korea

{wwj, wolee}@ee.knu.ac.kr, mhlee@knu.ac.kr

² Department of Information & Communication Engineering, Dongguk University,
707 Seokjang-Dong, Gyeongju, Gyeongbuk 780-714, Korea

swban@dongguk.ac.kr

³ Department of Electronic Engineering, Sogang University,

1 Shinsu-Dong, Mapo-Gu, Seoul 121-742, Korea

{min8328, hpark}@sogang.ac.kr

Abstract. In this paper, we propose a new multiple sensory fused human identification model for providing human augmented cognition. In the proposed model, both facial features and mel-frequency cepstral coefficients (MFCCs) are considered as visual features and auditory features for identifying a human, respectively. As well, an adaboosting model identifies a human using the integrated sensory features of both visual and auditory features. In the proposed model, facial form features are obtained from the principal component analysis (PCA) of a human's face area localized by an Adaboost algorithm in conjunction with a skin color preferable attention model. Moreover, MFCCs are extracted from human speech. Thus, the proposed multiple sensory integration model is aimed to enhance the performance of human identification by considering both visual and auditory complementarily working under partly distorted sensory environments. A human augmented cognition system with the proposed human identification model is implemented as a goggle type, on which it presents information such as unknown people's profile based on human identification. Experimental results show that the proposed model can plausibly conduct human identification in an indoor meeting situation.

Keywords: human augmented cognition, human identification, multiple sensory integration model, visual and auditory, adaptive boosting, selective attention.

1 Introduction

Human augmented cognition is one of the topics of cognitive science to extend a user's abilities via computational technologies. A person, even if he or she is not handicapped, has bottlenecks, limitations and biases in cognition. For example, limitations in attention, memory, learning, comprehension, visualization abilities, and decision making. The goal of human augmented cognition research is to develop

computational methods and tools to overcome these problems and to improve human cognition abilities.

Over the last couple of decades, there has been a lot of interesting on the design and development of several assistive devices aiming to provide people with visual impairments with ability of device manipulation in their daily activities. Most of these devices have been focused on enhancing the interaction with machines and environments of a user who is blind or visually impaired in dealing with a computer monitor, a personal digital assistant, a cellular phone, and indicating road traffic signals [1, 2]. Although these efforts are very essential for the quality of life of those visually handicapped people, such an assistant system is also helpful on the purpose of augmented cognition for common people to enlarge his or her cognition ability when they confront complex and distraction situation.

Thus, recently, the human augmented cognition systems such as visual and auditory assistance systems have received more attention from many smart-electronic device communities [3]. In order to implement those assistive systems, human identification technologies are one of important issues. In terms of human identification, face detection and recognition researches have been tremendously conducted as much as an amount of its importance [4, 5]. However, those face recognition researches have been utilized only visual information of face in order to identify human, which have troubles caused by various factors such as illumination change, image affine transform, distortion, and occlusion in real situation [4, 5]. Moreover, even though many researchers have proposed only auditory information based speaker detection and recognition, these models also have difficulties caused by various sound distortion occurred in real complex environment until now [6].

For solving these problems, some researchers have been proposed a combined visual-auditory approach considering both visual property and auditory property for human identity recognition [7, 8]. However, these models are considering different sensory features in a concatenating manner but an integrating manner. Therefore, those human identification systems do not consider associated features that may provide more complicate information for enhancing human recognition.

Thus, in this paper, we proposed a new visual-auditory fused model using an integrated manner of multiple sensory features for enhancing human identification. In order to obtain visual-auditory features, firstly visual and auditory features are extracted from face and speech of a human. Facial form features are extracted as visual features from principal component analysis(PCA) of the facial area localized an Adaboost algorithm in conjunction with a skin color preferable attention model [9-12]. Also MFCC features are extracted as auditory features from voice of a human. Then, the extracted visual and auditory features are integrated, which are used as input of a human identification model implemented by a sensory fusion adaboosting model [13]. The proposed human identification model is adapted to a goggle type human augmented cognition system, which provides information such as unknown people's profile through a goggle lens type screen.

This paper is organized as follows; Section 2 describes the proposed multiple sensory integrated model for human identification. The implemented goggle based human augmented system and experimental results will be followed in Section 3. Section 4 presents our conclusions and discussions.

2 Proposed Multiple Sensory Integration Model

Fig. 1 shows the proposed multiple sensory integration model. In order to robustly extract face features from visual information, it needs to consider more robust face detection. In this paper, we consider skin color preferable selective attention model which is to localize a face candidate. The proposed face detection method has smaller computational time and lower false positive detection rate than the well-known Adaboost face detection algorithm.

In order to robustly localize candidate regions for faces, we make skin color intensified saliency map (SM) which is constructed by selective attention model reflecting skin color characteristics. Figure 1 shows the skin color preferable saliency map model, in which red(r), green(g), blue(b) color features are extracted from input image. Intensity feature is generated by integrating the skin color filtered red(r), green(g), blue(b) color features. R-G color opponent feature is obtained from red(r) and green(g) color features and edge feature is generated using R-G color opponent feature. Then, the intensity, edge, and color opponent feature maps are constructed by the Gaussian pyramid processing and CSD&N algorithms [10]. Finally, a face color preferable SM is generated by integrating these three different feature maps, from which the face candidate regions are localized by applying a labeling based segmenting process [11]. The localized face candidate regions are subsequently categorized as final face candidates by the Haar-like form feature based Adaboost algorithms [11, 14]. As well, the visual features to be integrated with the auditory features are generated by projecting the localized face area on the selected principal components obtained from the principal component analysis (PCA) [12].

On the other hand, in order to extract low-level features of an input auditory signal, we consider mel-frequency cepstral coefficients (MFCC) feature extraction method which is commonly used in HMM Tool Kit (HTK) [15]. In the auditory feature extraction block, input signal is pre-emphasized through a first-order digital filter, whose transfer function is given as $1 - 0.97z^{-1}$ to make the signal spectrally more

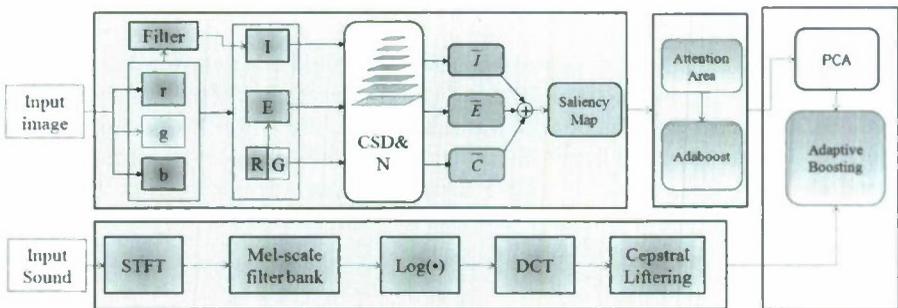


Fig. 1. Proposed multiple sensory integrated model for human augmented cognition; r :red, g : green, b :blue, Filter: skin color filter, I : intensity, E :edge, $R G$: normalized red-green color opponent, CSD&N: center surround difference and normalization algorithms, \bar{I} : intensity feature map, \bar{E} : edge feature map, \bar{C} : color opponent feature map, Adaboost: adaptive boost, STFT: short time Fourier transform, $\text{Log}(\bullet)$: logarithm, DCT: discrete cosine transform, PCA: principal components analysis.

flattened. Then, non-stationary speech signals are windowed by 25 ms-long Hamming window with a frame rate of 10 ms for applying short-time Fourier transform (STFT). The magnitude spectrum of each frame is weighted and summed up to make 24 filterbank outputs according to mel-scalc. The outputs of mel-scale filterbank are logarithmically scaled and converted into 12-order MFCCs by discrete cosine transform (DCT). Then, these cepstral coefficients are liftered for pragmatic reasons and these are packed with log energy feature of the frame into 13-dimension feature vector.

Finally, the human identification is conducted by a multiple sensory fusion adaboosting model using integrated features of facial form features as visual features and MFCC as auditory features.

2.1 Visual Feature Extraction

Face detection is one of important keys to enhance the performance of human identification. Even though the conventional face detection models based on an Adaboost algorithms show good performance in real time environments, it still has troubled with false positive detection rate and heavy computational load in complex environment. In order to enhance those problems, we consider a localizing method for face candidate regions, which is based on skin color preferable attention model. The proposed method effectively reduces the region of interesting area in a complex input visual scene.

For localizing the face candidate areas, we consider the skin color filtered intensity, R·G color opponent, and its edge feature, which are used as inputs for skin color preferable attention model. Thus, after extracting r, g, and b color features from input color image, the intensity and normalized red(R) and green (G) color features are extracted, which are known to effect on reducing influence of luminance like human visual system do [11].

The skin color filtered intensity feature is extracted from R, G, and B satisfying the dedicated ranges of R, G, and B shown in the following rules in Eq. (1)[11].

$$\begin{aligned} & r > 95, g > 40, b > 20 \text{ and} \\ & \max\{r, g, b\} - \min\{r, g, b\} > 15 \text{ and} \\ & |r - g| > 15 \text{ and } r > g \text{ and } r > b \end{aligned} \quad (1)$$

As a previous work, the R·G color opponent feature has been shown that it plays a more robust contribution factor to discriminate characteristics between face and non-face area than other color opponent features [9-11]. Therefore, R·G color opponent feature is considered one of face color preferable features. In order to enhance of edge magnitude for candidate face areas in a complex scene, we also consider the edge of R·G color opponent feature which is construed by Eq. (2) and the sobel edge operator is applied as an edge operator [16].

$$R \cdot G = |R - G| \quad (2)$$

Then, we consider the on-center and off-surround operation by the Gaussian pyramid images with different scales from 0 to n -th level whereby each level is made by the sub-sampling of 2^n , thus it is able to construct 3 feature bases such as intensity (I), and the edge (E), and color (R·G). Then, the center-surround features are constructed by the difference operation between the fine and coarse scales in the Gaussian pyramid

images [11]. Consequently, the three feature maps such as \bar{I} , \bar{E} , and \bar{C} , where stand for intensity, edge, and R-G color opponency, can be obtained by the center-surround difference algorithm [11].

A saliency map (SM) is constructed by the normalized summation of those three feature maps as shown in Eq. (3).

$$SM = \text{Norm}(\bar{I} + \bar{E} + \bar{C}) \quad (3)$$

After the face candidate areas are segmented by a labeling process for binarized saliency map, which is obtained by the Otsu's threshold, the localized face candidate areas are used as input of the Adaboost algorithms for verifying the face regions [16, 14]. Finally, the PCA extracts facial features for recognizing human faces [12].

2.2 Auditory Feature Extraction

Auditory features contained in a speech signal are able to be categorized into three kinds; linguistic message, speaker information, and acoustic channel characteristics. In order to extract only features reinforcing human recognition, we need to separate speaker information from others as much as possible. According to speech synthesis model, linear prediction (LP) coefficients can be good features well representing vocal tract excitation which is valuable speaker information. But in practice, MFCC feature extraction method works well in adverse condition as well as in normal condition. Moreover, MFCC features are good at both speaker recognition system and speech recognition system with less computation complexity, so we use MFCC features for human identification [6].

MFCC features are usually combined with additional features such as the first- and second-order delta features which reduce the word recognition error. In the proposed multiple sensory integration model, these additional delta coefficients are not only insignificant for speaker discrimination in the aspect of perception but also likely to cause worse recognition results as well-known paradox, 'curse of dimensionality' [17]. Hence, we do not use first- and second order delta coefficients.

Input sound samples are pre-emphasized through a first-order digital filter whose transfer function is given as $1 - 0.97z^{-1}$. This filter has almost linearly increasing frequency response, so speech signals become spectrally more flattened after filtering. Then, using a Hamming window which is given by Eq.(4), speech signals are split into 25 ms short-time segments called as frames that are windowed at every 10 ms time advance.

$$\text{win}(n) = 0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right), 0 \leq n \leq N-1 \quad (4)$$

Each frame is zero-padded, and taken short-time Fourier transform (STFT) makes its spectrum. The magnitude values of the spectrum are weighted by 24 triangular windows that are centered at each of mel-scale frequency points half-overlapping with adjacent bands and summed up to make 24 band outputs. Typical mel-scale is given by Eq. (5).

$$\text{mcl}(f) = 2595 \log_{10}\left(1 + \frac{f}{700}\right) \quad (5)$$

From the above procedure, the k th-band output at frame t , $o_k(t)$, which is scaled logarithmically to produce a log-spectral parameter [18], is represented as Eq.(6), where s_k and e_k denote the start and end points of the k th-band, respectively.

$$o_k(t) = \log \left(\sum_{f=s_k}^{e_k} \omega_k(f) |S_\omega(f,t)| \right) \quad (6)$$

f denotes the frequency index, $|S_\omega(f,t)|$ represents the magnitude spectral value at f and t , and $\omega_k(f)$ is a weight function corresponding to a triangular window of the k th-band.

Then, MFCCs are obtained by taking the DCT to the log-spectral parameters. The i th coefficient of M -order MFCCs at frame t , $c_i(t)$, is expressed as Eq.(7) [19], where K denotes the number of mel-scaled bands.

$$c_i(t) = \sum_{k=1}^K o_k(t) \cos \left(\frac{i(k-0.5)\pi}{K} \right) \quad (7)$$

The principal advantage of the cepstral coefficients is that they are generally decorrelated and allow a diagonal covariance to be used in a classifier. One minor problem is that higher order cepstra are numerically very small and this results in related parameters such as covariances having a wide range. Actually, it does not affect to performance of a classifier, but for pragmatic reasons such as storing data in limited precision, displaying parameters, etc., we re-scale the cepstral coefficients to have similar magnitudes by following Eq. (8), where L denotes the liftering parameter.

$$\hat{c}_i(t) = \left(1 + \frac{L}{2} \sin \frac{\pi i}{L} \right) c_i(t) \quad (8)$$

2.3 Adaptive Boosting for People Recognition

As we mentioned above, we obtain visual features from each face image using the PCA algorithm and extract auditory features by MFCC feature extraction method. After feature extraction, we need to construct a classifier to identify the person. In the proposed multiple sensory integration model, we use the Adaboost algorithm to integrate visual and auditory features and identify a human. The Adaptive boosting algorithm generally integrates the weak classifier's results and uses the weighted voting method to construct a strong classifier as shown in Eq. (9) [13].

$$H(x) = \text{sign} \left(\sum_{t=1}^T \alpha_t h_t(x) \right)$$

(9)

x : Input
 H : Final hypothesis
 α_t : Weight for weak classifier
 h_t : Weak classifier

The classification result of each simple classifier, $h_i(x)$, has the value of 1 or -1. And the weight, $\alpha_i(x)$ is calculated by Eq. (10).

$$\alpha_i = \frac{1}{2} \ln \left(\frac{1 - \varepsilon_i}{\varepsilon_i} \right) \quad (10)$$

The weight becomes greater when error rate, ε_i , is smaller value. Therefore, the more accurate classifier, the higher weight it has. In order to build a strong classifier, $H(x)$, we need many weak classifiers. We need 290 weak classifiers since the sizes of extracted features are 160 and 130 for visual and auditory, respectively. Actually, the lengths of auditory features are different from their lengths of raw data. If someone pronounced a word during long time, the length of feature is long, or during short time, the length of features is short. Therefore we need to make them to have the same length. In order to do, we divide the auditory features to 10 sections and use average value of each section. Finally we obtain 130 auditory features because each section consists of 13 values.

According to the Adaboost algorithm, weak classifiers will be accepted if they have an error rate with below 50% [13]. Therefore we build each weak classifier with a single threshold. Initial threshold is set by median value of feature in positive group and average value of feature in negative group. Next, we modify the threshold in the range of -50% and +50% by increasing 1%. A threshold that has minimum error rate is determined as a final threshold. Every weight and threshold is calculated in learning process, and recognition process is performed using these parameters.

3 Experimental Results

3.1 Hardware Platform and Scenario for Experiment

For the experiment, we have developed a goggle based human augmented system for supporting user to provide information about unknown or not memorized participant, by presenting contexts on the screen in a meeting and conference situation. As shown in Fig. 2, the system have equipped with 2-phinhole CCD camera with 2 Microrobot's USB image grabbers for recognizing scene and user gaze, and 2 TCM100 microphone with Terra Tech's 6fire USB amplifier for localizing and recognizing auditory source signal. And, we make a meeting scenario for demonstrating the performance of the proposed system which is sequentially consisted of entrance and introduction of participant, introduction of participant after participant's seating down, discussion and presentation, and decision and conclusion of meeting. Then, we took 14 videos in a different illumination environment varying from 140 to 412 lux with three people who are participants in a meeting, through which we obtained visual and auditory database for experiments and verification of the developed system under the provided scenario.

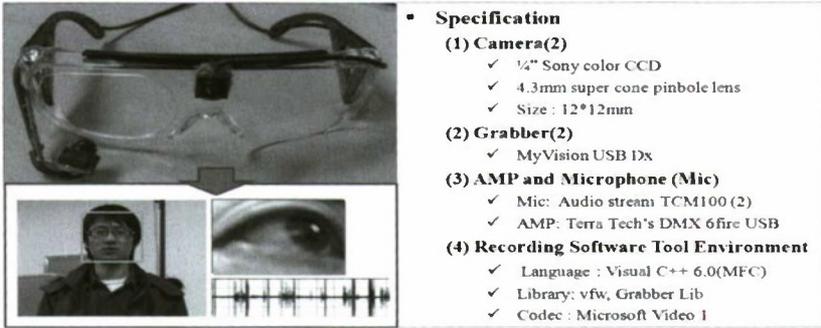


Fig. 2. Developed goggle based augmented cognition platform based on visual and auditory sensory integrated human identification

3.2 Experimental Results

In the face detection experiment, we captured 420 images from 14 videos for introduction of participant when entering and introduction of participant after seating down in a meeting place.

For speech acquisition, the sampling frequency of A/D converter for input speech signal is typically 16 kHz and A/D precision is 16 bits. In practice, a 16 kHz sampling rate is sufficient for the speech bandwidth (8 kHz). It is empirically verified that the word recognition error stops being reduced when the sampling rate is increased over 16 kHz [17]. Therefore, parameters for an A/D converter are determined as a 16 kHz sampling rate and 16-bits precision. In frequency analysis, we set short-time segment width to 25 ms as a compromise between the stationarity assumption and the frequency resolution. Also, we set frame shift to 10 ms which is typically used. Related to these parameters, 25 ms length under a 16 kHz sampling rate corresponds to 400 samples in a frame, so 512 FFT points are automatically computed to generate a spectrum. The number of channels in the mel-scale filterbank is 24 because too many channels may cause unwanted fluctuation on the spectral envelop, and too few channels may smooth details [20]. Finally, the dimensionality of a parametric vector is reduced to 12 again by taking DCT to convert log-spectral parameters into MFCCs. These features form a 13-order feature vector with an additional log-energy feature for the current frame. As a result, a 13-order auditory feature vector is supplied for adaptive boosting module at every 10 ms.

Moreover, in order to evaluate our multiple sensory integration model, we use 54 multiple sensory data, consisting of 54 face images and 54 speech data, obtained from 14 video database for three people. Half of the dataset is used for training and the others are used for evaluation. The adaboosting model for human identification is based on a two-class classifier. Thus, among 27 data for three persons, 9 data for one person are used as a positive set and 18 data for the other two persons are considered as a negative set and this classification experiment is repeatedly conducted for 3 times by changing positive and negative datasets in a combination manner.

Fig. 3 shows the experimental results for every process of localizing face areas in the proposed face preferable selective attention model. Because the proposed face preferable selective attention model was considered skin color filtered intensity and

R-G color opponent, and edge of R-G color opponent, face color regions are naturally more intensified than other background in a saliency map. Therefore, candidate face areas are simply localized through a labeling based segmenting process with a size constraint in a binarized saliency map. As a result, this processing can effect on enhancing computational load and false-positive detection rate for the Adaboost face detection model as shown in Fig. 3. Table 1 also shows the performance of the proposed face detection for our face database. Even though the correct detection rate of the proposed model is slightly lower than that of the conventional Adaboost face detector in varying illumination environments, the proposed model shows better results for the false positive detection rate as shown in Table 1. Moreover, as a previous work, we have shown that the proposed face detection model not only shows a good performance but also enhances the computational load and the false positive detection rate for an open database [11].

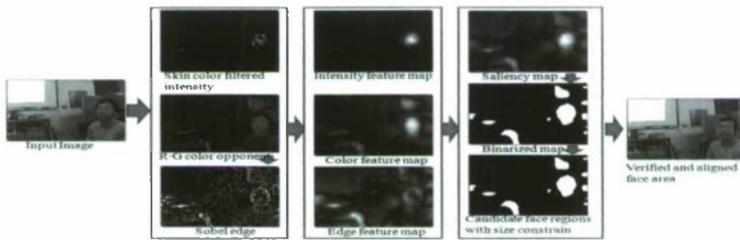


Fig. 3. Experimental results of the proposed face detection model

Fig. 4 shows the human identification performance of the proposed multiple sensory integration model. Through this experiment, we aimed to show that the human identification performance can be enhanced much by considering sound information together with visual information than only considering visual information. As shown in Fig. 4, when we considered only visual information, the performance is low as 79.0%. However, the performance becomes very much higher as 98.9% by considering both sound information and visual information than 79.0% by considering only visual information, which shows that the proposed multiple sensory integration model can plausibly enhance the human identification performance. On the other hand, in this experiment, we obtained 100% human identification performance when we considered only sound information, which has been caused by considering only clear sound data in the experiments. However, in the case of the image data involved in the experiments, the image database was obtained under varying illumination conditions. Thus, we have verified the performance of the proposed multiple sensory integration model by showing that the human identification performance can be enhanced by additionally considering sound features as well as visual features in an integrated manner.

Table 1. Comparison of face detection performance

	Proposed detection model	Conventional Adaboost
True positive	96.2%(404/420)	98.3%(413/420)
False positive	4.5%(19/423)	11.2%(52/465)

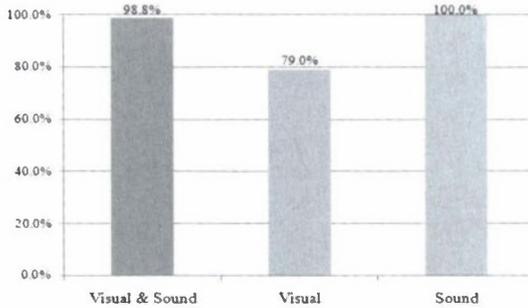


Fig. 4. The performance of multiple sensory integration based human identification model

4 Conclusion

We proposed an adaptive boosting based multiple sensory integration model for human identification by combining visual and auditory features. In order to extract visual features robustly, we consider not only the face preferable selective attention model for enhancing the computational load and false positive detection rate of Adaboost face detector, but also a PCA approach for extracting proper facial features as well as reducing dimension of facial features. In addition, we also considered the well-known MFCC for extracting auditory features robustly.

Even though the multiple sensory integration based human identification model has shown the plausible performance in the experiments using our database reflecting low illumination environment, it seems to be hard to deploy the proposed model in a real system since the proposed model is lack of incremental mechanism in online feature extraction and learning concept. Therefore, as a further work, we consider online increment learning concept for developing a more advanced human augmented cognition system. Moreover, we are considering more experiments using noisy sound data in order to verify the performance of the proposed model.

Acknowledgments. This research was supported by the Converging Research Center Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2009-0082262).

References

1. Balasubramanian, V., et al.: Human-Centered Machine Learning in a Social Interaction Assistant for Individuals with Visual Impairments. In: Symposium on Assistive Machine Learning for People with Disabilities at NIPS (2008)
2. Velázquez, V., Maingreud, F., Pissaloux, E.: Intelligent Glasses: A New Man-Machine Interface Concept Integrating Computer Vision and Human Tactile Perception. In: EuroHaptics 2003 (2003)
3. Schmorow, D.D., Kruse, A.A.: DARPA's Augmented Cognition Program-tomorrow's human computer interaction from vision to reality: building cognitively aware computational systems. In: IEEE Human Factors and Power Plants, pp. 7.1–7.4. IEEE Press, Los Alamitos (2002)

4. Andrea, F.A., Michele, N., Daniel, R., Gabriele, S.: 2D and 3D face recognition: A survey. *J. Pattern Recognition Letters* 28, 1885–1906 (2007)
5. Tolba, A.S., El-Baz, A.H., El-Harby, A.A.: Face Recognition: A Literature Review. *J. Signal Processing* 2, 88–103 (2006)
6. Campbell, J.P.: Speaker recognition: A tutorial. *Proc. of the IEEE* 85(9), 1437–1462 (1997)
7. Feng, W., Xie, L., Zeng, J., Liu, Z.Q.: Audio-visual human recognition using semi-supervised spectral learning and hidden Markov models. *J. Visual Languages and Computing Vision* 20, 188–195 (2009)
8. Albiol, A., Torres, L., Delp, E.J.: Fully automatic face recognition system using a combined audio-visual approach. *IEE Proc. Vis. Image Signal Process* 152, 318–326 (2005)
9. Won, W.J., Yeo, J., Ban, S.W., Lee, M.: Biologically Motivated Incremental Object Perception Based on Selective Attention. *J. Pattern Recognition and Artificial Intelligence* 21(8), 1293–1305 (2008)
10. Won, W.J., Jang, Y.M., Ban, S.W., Lee, M.: Biologically Motivated Face Selective Attention Model. In: Ishikawa, M., Doya, K., Miyamoto, H., Yamakawa, T. (eds.) *ICONIP 2007, Part I. LNCS*, vol. 4984, pp. 953–962. Springer, Heidelberg (2008)
11. Kim, B., Ban, S.W., Lee, M.: Improving AdaBoost Based Face Detection Using Face-Color Preferable Selective Attention. In: Fyfe, C., Kim, D., Lee, S.-Y., Yin, H. (eds.) *IDEAL 2008. LNCS*, vol. 5326, pp. 88–95. Springer, Heidelberg (2008)
12. Smith, L.: A Tutorial on Principal Components Analysis.
http://www.cs.otago.ac.nz/cosc453/student_tutorials/principal_components.pdf
13. Freund, Y., Schapire, R.E.: A short introduction to boosting. *J. Japanese Society for Artificial Intelligence*. 14(5), 771–780 (1999)
14. Viola, P., Jones, M.J.: Robust real-time face detection. *J. Computer Vision* 57(2), 137–154 (2007)
15. Young, S., Evermann, G., et al.: *The HTK Book (for HTK Version 3.4)*. Cambridge University Engineering Department, Cambridge (2009)
16. Gonzalez, R.C., Woods, R.E.: *Digital Image Processing*, 2nd edn. Prentice-Hall, Englewood Cliffs (2001)
17. Huang, X., Acero, A., Hon, H.W.: *Spoken Language Processing: a guide to theory, algorithm, and system development*. Prentice Hall PTR, Englewood Cliffs (2001)
18. Park, H.M.: *Adaptive Filtering Methods for Acoustic Noise Reduction and Noisy Speech Recognition*. In: Doctor's thesis, Department of Electrical Engineering and Computer Science, Division of Electrical Engineering, Korea Advanced Institute of Science and Technology (2003)
19. Davis, S.B., Mermelstein, P.: Comparison of parametric representations for mono-syllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 28(4), 357–366 (1980)
20. Gold, B., Morgan, N.: *Speech and Audio Signal Processing*. John Wiley & Sons, Inc., Chichester (2000)

Steady-State Genetic Algorithms for Growing Topological Mapping and Localization

Jinsok Woo¹, Naoyuki Kubota¹, and Beom-Hee Lee²

¹ Tokyo Metropolitan University

6-6 Asahigaoka, Hino, Tokyo 191-0065, Japan

² School of Electrical Engineering and Computer Science,
Seoul National University, Seoul, Korea

woo-jinseok@sds.tmu.ac.jp, kubota@tmu.ac.jp,
bhlee@snu.ac.kr

Abstract. This paper proposes a method of simultaneous localization and mapping based on computational intelligence for a robot partner in unknown environments. First, we propose a method of topological map building based on a growing neural network. Next, we propose a method of localization based on steady-state genetic algorithm. Finally, we discuss the effectiveness of the proposed methods through several experimental results.

Keywords: Simultaneous Localization and Mapping, Informationally Structured Space, Mobile Robots, Neural Networks, Genetic Algorithm.

1 Introduction

Recently, robots have been familiar for people, and we expect human-friendly robots co-existing in human living environments. Such a robot needs various capabilities such as learning, inference, and prediction for human interaction, and such capabilities are interconnected each other in the total system. In the previous works, multi-strategic learning has been discussed to integrate multiple inference types and/or computational mechanisms in one learning system [1], e.g., integration of symbolic and numerical learning, a hybrid computation of discrete space and continuous space, integration of stochastic search and deterministic heuristic search, and others. A multi-strategic approach of path planning and behavioral learning [2], a reinforcement learning based on value iteration and policy iteration, and others have been proposed in the field of intelligent robotics.

A human-friendly robot should have an environmental map for co-existing with people, but it is very difficult to build the environmental map beforehand. The important functions are to build up an environmental map and to estimate and correct the self-location. The robot builds up an environmental map according to the position and posture of the robot, while the robot estimates the position and posture according to the built environmental map. This is a mutual nesting structure, and is well known as a simultaneous localization and mapping (SLAM) [3-10]. SLAM is also considered as one of the multi-strategic learning methods. Map building by mobile robots has a

long history [11-18]. There are two main methods of metric approach and topological approach. In the metric approach, an environment is represented by finite discrete space or a set of polygons. For example, in a cell decomposition method, a two-dimensional workspace is often divided into $M \times N$ rectangular cells. In the topological approaches, an environment is represented by a list of connectivities of places. Skeletonization methods directly generate intermediate points and paths, while the cell decomposition methods generate collision-free space. In the skeletonization methods, collision-free paths are basically generated according to polygonal objects approximated in a workspace. Visibility graph consists of edges connecting visible pairs of vertices of the polygonal objects. In the visibility graph, the shortest path between two points can be generated easily by selecting edges. However, it is dangerous for a mobile robot to move along the generated path, because the path is adjacent to the vertices of the polygonal objects. To overcome this problem, a Maklink graph can be used to generate a safe path. This method can be considered as one of the approximated Voronoi diagrams. In the Maklink graph, a candidate point is represented as a middle point between two vertices, and a path is generated by connecting some intermediate points. Although the generated path is safe, it might not be the shortest.

Next, we explain the background of the localization method for mobile robots. Kalman filters have been applied for the localization in case of small and incremental dead-reckoning errors, and multi-hypothesis Kalman filters have been applied for the localization based on beliefs using the mixture of Gaussians. Furthermore, Monte Carlo localization has been applied for the localization [10]. Monte Carlo localization uses the belief by a set of samples called particles, and this method is known as a particle filter. The particle filter is one of non-parametric Bayesian filtering methods. The particle filter can approximately represent the posteriors by a random collection of weighted particles of the desired distribution. As the number of samples becomes very large, this Monte Carlo characterization becomes an equivalent representation to the usual functional description of the posterior probability density function, and the sequential importance sampling filter can approach the optimal Bayesian estimate. However, the particle filter takes much computational time and cost.

In our previous works [19], we used image processing to estimate the self-location of the robot based on the cell decomposition method, but it is very difficult to deal with the environmental lighting conditions. Furthermore, the accuracy of the map building depends strongly the granularity of the map. Therefore, we proposed a topological map building method based on a growing neural network as a topological approach[4]. A growing neural network can add neurons and their connections to the network. Furthermore, we applied a steady-state GA (SSGA) to update the estimated the self-position of the robot by using the measured distance and topological map.[4] The proposed method was applied to SLAM in a city hall, a parking area, and university cafeteria, and we compared the experimental result of the proposed method with that of particle filter[21]. However, the developed mobile robot is too large to use as a robot partner at home. Therefore, we develop a small size of robot partner, and apply the proposed method to a living room in this paper. Next, we discuss the effectiveness of the proposed method through experimental results.

This paper is organized as follows. Section 2 explains the hardware and control method of the mobile robot. Section 3 proposes a method for topological map building

based on growing neural network and steady-state genetic algorithm. Section 4 shows that the robot can perform simultaneous localization and mapping by using the proposal method.

2 SLAM for Informationally Structured Space

2.1 Informationally Structured Space

Recently, various types of remote observing systems of elderly people living alone in a house have been developed for the detection of their emergency as the population of elderly people increases. The introduction of coexisting human-friendly robot partners are one of possible solutions to realize the remote observation of elderly people. Wireless sensor networks realize to gather the huge data on environments for remote monitoring. However, it is very difficult to store all of huge data in real time. Furthermore, some features should be extracted from the gathered data to obtain the required information. The accessibility within environmental information is essential for both people and robots. Therefore, the environment surrounding people and robots should have a structured platform for gathering, storing, transforming, and providing information. Such an environment is called informationally structured space (Fig.1). The structuralization of informationally structured space realizes the quick update and access of valuable and useful information for users. If the robot can share the environmental information with people, the communication with people might become very smooth and natural.

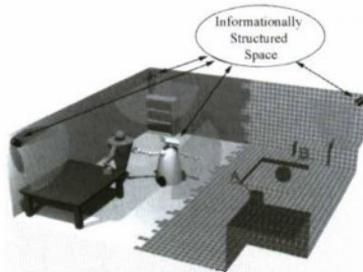
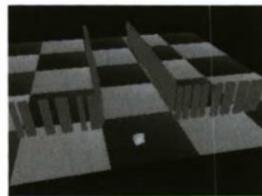


Fig. 1. The concept of informationally structured space



(a) A real environment



(b) An environmental map based on SLAM

Fig. 2. An example of SLAM

A robot partner needs the intelligent capabilities of environmental map building and physical behavioral learning in a real world at least. The map building of the co-existing environment is performed by SLAM. Figure 2 shows the environment and its corresponding map as a result of SLAM. If a robot partner automatically performs SLAM, it is easy to perform the remote monitoring.

2.2 A Robot Partner: MOBiMac

We developed a partner robot; MOBiMac shown in Fig.3. Two CPUs are used for the interaction with a human and the control of the robotic behaviors. The robot has two servo motors, eight ultrasonic sensors, a laser range finder (LRF) and a CCD camera. An ultrasonic sensor can measure the distance to objects. The LRF can measure the distances up to approximately 4,095 mm in 682 different directions where the covering measurement range is 240° . Therefore, the robot can take various actions such as collision avoidance, human tracking, and line tracing. The behavior modes of this robot are human detection, human communication, behavior learning, and behavioral interaction. The communication with a person is performed by utterance as the result of voice recognition and gestures as the result of human motion recognition.

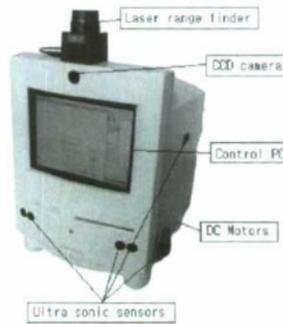


Fig. 3. Robot Partner; MOBiMac

Various intelligent methods for mobile robots have been proposed such as production rules, Bayesian networks, neural networks, fuzzy inference systems, and classifier systems. We have applied fuzzy inference systems to represent behavior rules of mobile robots, because the behavioral rules can be designed easily and intuitively by human linguistic representations. A behavior of the robot can be represented using fuzzy rules based on simplified fuzzy inference. In general, a fuzzy if-then rule is described as follows,

If x_1 is $A_{i,1}$ and ... and x_M is $A_{i,M}$
Then y_1 is $w_{i,1}$ and ... and y_N is $w_{i,N}$

where $A_{i,j}$ and $w_{i,k}$ are the Gaussian membership function for the j th input and the singleton for the k th output of the i th rule; M and N are the numbers of inputs and outputs, respectively. Fuzzy inference is performed by,

$$\mu_{A_{i,j}}(x_j) = \exp\left(-\frac{(x_j - a_{i,j})^2}{b_{i,j}^2}\right) \tag{1}$$

$$\mu_i = \prod_{j=1}^M \mu_{A_{i,j}}(x_j) \tag{2}$$

$$y_k = \frac{\sum_{j=1}^R \mu_j w_{i,k}}{\sum_{j=1}^R \mu_j} \tag{3}$$

where $a_{i,j}$ and $b_{i,j}$ are the central value and the width of the membership function $A_{i,j}$; R is the number of rules. Outputs of the robot are output levels of the left and right motors ($N=2$). Fuzzy controller is used for collision avoidance and target tracing behaviors. The inputs to the fuzzy controller for collision avoidance are the measured distance to the obstacle by LRF ($M_c=8$). The number of directions of LRF is reduced in 8 by choosing the minimal distance in each sensing range. The inputs to the fuzzy controller for target tracing are the estimated distance to the target point and the relative angle to the target point from the moving direction ($M_t=2$).

In general, a mobile robot has a set of behaviors for achieving various objectives, and must integrate these behaviors according to the environmental conditions. Therefore, we proposed the method for multi-objective behavior coordination. The multi-objective behavior coordination can integrate outputs of several behaviors according to the time-series of perceptual information, while the original subsumption architecture selects one behavior. This multi-objective behavior coordination is composed of a sensory network, behavior coordinator, and behavior weight updater. The sensory network extracts perceptual information based on sensing data and updates the parameters of sensors recursively according to the perceptual information. A behavior weight is assigned to each behavior. Based on eq.(3), the output is calculated by

$$y_k = \frac{\sum_{j=1}^K wgt_j(t) \cdot y_{j,k}}{\sum_{j=1}^K wgt_j(t)} \tag{4}$$

where K is the number of behaviors; $wgt_j(t)$ is a behavior weight of the j th behavior over the discrete time step t . By updating the behavior weights, the robot can take a multi-objective behavior according to the time series of perceptual information. The update amount of each behavior is calculated as follows,

$$\begin{bmatrix} \Delta wgt_1 \\ \Delta wgt_2 \\ \vdots \\ \Delta wgt_K \end{bmatrix} = \begin{bmatrix} dw_{1,1} & dw_{1,2} & \cdots & dw_{1,L} \\ dw_{2,1} & & & \\ \vdots & & \ddots & \vdots \\ dw_{K,1} & & \cdots & dw_{K,L} \end{bmatrix} \begin{bmatrix} si_1 \\ si_2 \\ \vdots \\ si_L \end{bmatrix} \tag{5}$$

where si_i is the parameter on the perceptual information; L is the number of perceptual inputs. This method can be considered as a mixture of experts if the behavior coordinator is considered as a gating network.

3 Simultaneous Localization and Mapping

3.1 Growing Topological Map Building

Map building can be regarded as one of unsupervised learning approaches where sampling data are noisy and imprecise, because the measurement noise is included in the measured data (Fig.4). Self-organizing map (SOM) is often applied for extracting a relationship among measured data, since SOM can learn the hidden topological structure from the data [22]. The original SOM used the pre-defined number of nodes. Neural gas has been also used for constructing a topological map, and furthermore, growing neural gas is used for incremental learning of the topological structure [23-27]. Local error measures are used for determining the place to insert new nodes. The competitive Hebbian rule generates the edges between nodes.

The addition of nodes and the generation of the edges between nodes can be applied to topological map building (Fig.5). Therefore, we proposed a topological

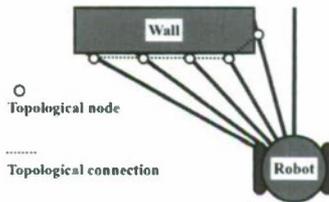


Fig. 4. Topological map building

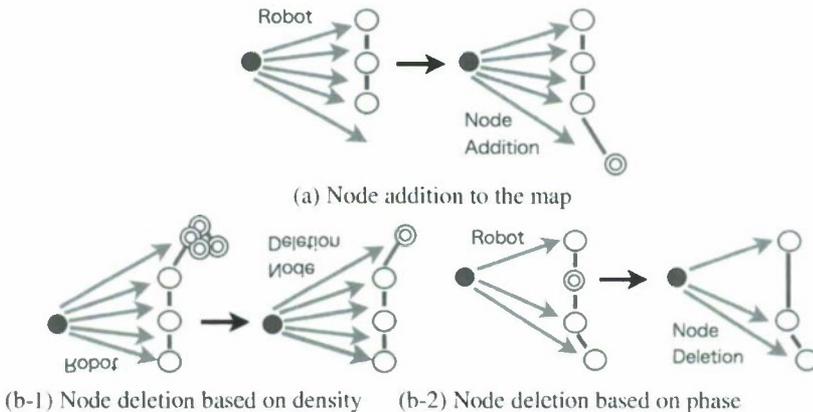


Fig. 5. Growing topological Mapping Building

map building method based on the concept of growing neural gas. We explain the method in the following. At the first measurement, the measurement points are added as the initial nodes of the topological map. Afterward, the topological map is updated according to the measured data.

When the i th reference vector of the topological map is represented by \mathbf{r}_i , the Euclidean distance between an input vector and the i th reference vector is defined as

$$d_i = \|\mathbf{V} - \mathbf{r}_i\| \quad (6)$$

Where $\mathbf{r}_i = (r_{1,i}, r_{2,i}, \dots, r_{N,i})$. Next, the k th output node minimizing the distance d_i is selected by

$$k = \operatorname{argmin}\{\|\mathbf{V} - \mathbf{r}_i\|\} \quad (7)$$

The selected output node is the nearest point on the topological environmental map according to the measured distance. Furthermore, the reference vector of the i th output node is trained by

$$\mathbf{r}_i \leftarrow \mathbf{r}_i + \xi \cdot \zeta_{k,i} \cdot (\mathbf{V} - \mathbf{r}_i) \quad (8)$$

where ξ is a learning rate ($0 < \xi < 1.0$); $\zeta_{k,i}$ is a neighborhood function ($0 < \zeta_{k,i} < 1.0$).

The number of nodes, n_{node} is gradually increased when there is no node corresponding to input data. The number of inputs in each sampling of the distance information is L ($L=682$). We show the procedure of the topological map building:

Step 1: Initialization of the map based on the first measurement; $t=1$.

Step 2: Distance measurement ($\mathbf{z}(t)$) and Motion Output ($\mathbf{y}(t)$)

Step 3: for $i=1$ to L do

Step 4: Select k th node according to the distance d_k

Step 5: if $d_k > D_{max}$ then $n_{node}++$; add $\mathbf{r}_{n_{node}}$
otherwise, update \mathbf{r}_k

Step 6: end i

Step 7: Generate a set \mathbf{O}_k composed of near nodes with respect to the k th node.

Step 8: if the number of nodes in \mathbf{O}_k is larger than the predefined number n_{max} then
the least selected node is removed from the topological map.

Step 9: Remove unnecessary nodes

Step 10: $t++$

Step 11: go to step 2

This method is composed of three steps of node addition (*Step 5*), learning (*Step 5*), and node deletion (*Step 7-9*). The node deletion is performed in order to remove unnecessary and crowded nodes.

Figure 5 shows an example of topological map building. If the node does not exist in the position corresponding to the measured distance, a node is added to the map (Fig.5 (a)). If there are many nodes crowded, some of them are removed from the map (Fig.5 (b)).

3.2 Steady-State Genetic Algorithm for Localization

As one stream of evolutionary computing, genetic algorithms (GAs) have been effectively used for solving optimization problems in robotics [28-32]. GAs can produce a feasible solution, not necessarily an optimal one, with less computational cost. SSGA simulates the continuous model of the generation, which eliminates and generates a few individuals in a generation (iteration). A candidate solution called an individual is composed of numerical parameters of the revised values to the current position $(g_{i,1} \ g_{i,2})$ and rotation $(g_{i,3})$. In SSGA, only a few existing solutions are replaced by new candidate solutions generated by genetic operators in each generation [32]. In this paper, the worst candidate solution are eliminated and replaced with the candidate solution generated by the crossover and mutation. We use the elitist crossover and adaptive mutation. Elitist crossover randomly selects one individual and generates an individual by incorporating genetic information from the selected individual and best individual in order to obtain feasible solutions rapidly. Next, the following adaptive mutation is performed to the generated individual,

$$g_{i,j} \leftarrow g_{i,j} + \left(\alpha_j \cdot \frac{f_i - f_{\min}}{f_{\max} - f_{\min}} + \beta_j \right) \cdot N(0,1) \tag{9}$$

where f_i is the fitness value of the i th individual, f_{\max} and f_{\min} are the maximum and minimum of fitness values in the population; $N(0,1)$ indicates a normal random value; α_j and β_j are the coefficient and offset, respectively. In the adaptive mutation, the variance of the normal random number is relatively changed according to the fitness values of the population. Fitness value is calculated by the following equation,

$$fit_i = \lambda_1 (g_{i,1}^2 + g_{i,2}^2) + \lambda_2 g_{i,3}^2 + \lambda_3 \sum_{k \in K} d_k \tag{10}$$

where d_k is the distance between the measured point and its nearest node in the topological map; λ_1 , λ_2 , and λ_3 are weight parameters for multi-objective optimization. These weight parameters are heuristically determined. Therefore, this problem results in the minimization problem. The population size is G , and the number of iteration times is T . We show the procedure of SSGA for the localization in the following;

- Step 1: Initialization of samples and importance factors; $t=1$.
- Step 2: Distance measurement ($z(t)$) and Motion Output ($y(t)$)
- Step 3: for $i=1$ to G do
- Step 4: Adaptive Mutation
- Step 5: Evaluation
- Step 6: end i
- Step 7: for $i=1$ to T do
- Step 8: Least Fitness Selection
- Step 9: Elitist Crossover
- Step 10: Adaptive Mutation
- Step 11: end i
- Step 12: Update the self-position according to the best individual

Step 13: $t++$;
Step 14: goto Step 2

In the adaptive mutation in Step 4, some individuals partially inherit genetic information from the previous population.

4 Experimental Result

This section shows experimental results of the proposed method. Figure 7 shows an environment of an elevator hall where the size of this area is approximately 11 x 15 [m] and there are many obstacles. The robot starts at the initial position in the lower right in Fig.7 (a), moves along the red line, and goes back the initial position. The maximal number of nodes is 1,000. The population size of SSGA is 50, and the evaluation times of SSGA is 500 including the evaluation of individuals in the initialization. Because the robot does not have rotary encoders for dead reckoning, the robot must perform localization according to the distance information measured by LRF. Figure 8 shows an experimental result of the proposed method. The features

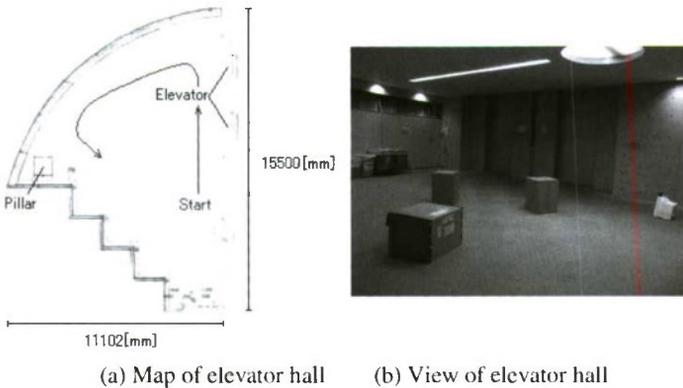


Fig. 7. An experimental environment (Case 1)

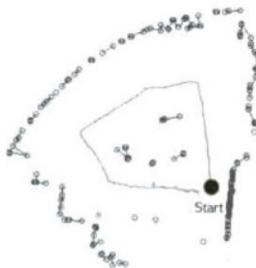


Fig. 8. An experimental result of SLAM (Case 1)

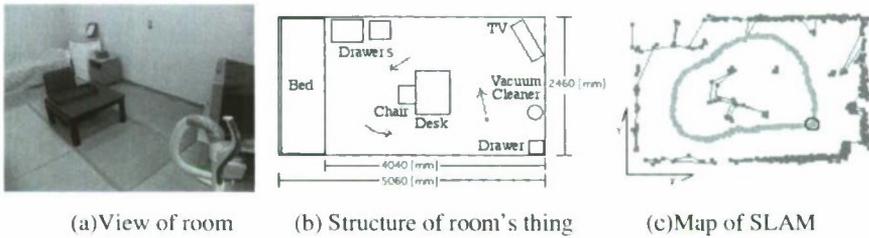


Fig. 9. Partner robot's SLAM In the room (Case 2)

of the environment are extracted by the proposed method, and the topological map is generated. In fact, the number of obstacles is 3 in the center of the obtained topological map. The final number of nodes is 361.

Figure 9 shows an example of a living room for elderly people where the size of this area is approximately 5×2.5 [m] and there are many obstacles (Case 2). Figure 11 shows an experimental result of the proposed method. The features of the environment are extracted by the proposed method, and the topological map is generated. Because the height of the table is nearly equal to the position of the LRF equipped with the robot in this result, the built map partially includes the top board of the table. Table 1 shows the number of nodes used in the map, the size of map, and the error of estimated position and posture in the final state of SLAM. The obtained result is efficient for the robot to conduct interaction with people.

Table 1. Status of the map obtained by SLAM in Case 2

Number of nodes	591
Measured distance (size of the room)	3977×2457 [mm]
Error of the estimated robot position	X:-45 [mm] Y:0.2 [mm]
Error of the estimated posture	2°

5 Summary

This paper discussed the SLAM of a mobile robot based on computational intelligence. We proposed the topological map building method for SLAM by using a growing neural network and a steady-state genetic algorithm for the localization. In the experiment results of the elevator hole in the university and a living room for elderly people, the map building was successfully done by the proposed method, although the rotary encoders for dead reckoning are not equipped with the robot and the moving direction of the mobile robot is not used in the proposed method. However, it is difficult to conduct map building using small objects such as legs of table and chairs in the living room, because such a leg is considered as a point, not a line or plane. Furthermore, SLAM might memorize a moving object as a static object. As a result, such a moving object can be noise in the map used for SLAM.

As a future work, we should consider the object property obtained from informationally structured space. We intend to perform experiments in the corridor in a large size of floor in order to show the effectiveness of the proposed method. Furthermore, we will develop a topological map building method based on the temporal reliability in a dynamic environment.

References

1. Michalski, R., Tecuci, G.: *Machine Learning: A Multistrategy Approach*. Morgan Kaufmann Publishing, San Francisco (1994)
2. Bianco, G., Cassinis, R.: Multi-strategic approach for robot path planning in an unknown environment (1993)
3. Thrun, S., Burgard, W., Fox, D.: *Probabilistic Robotics*. MIT Press, Cambridge (2005)
4. Sasaki, H., Kubota, N., Taniguchi, K.: Topological Map and Cell Space Map for SLAM of A Mobile Robot. *GESTS International Transactions on Computer and Engineering* 45(3) (2008)
5. Tomono, M., Yuta, S.: Object-based Localization and Mapping using Loop Constraints and Geometric Prior Knowledge. In: *Proc. of International Conference on Robotics and Automation*, pp. 862–867 (2003)
6. Thrun, S.: Robotic mapping: A survey. In: Lakemeyer, G., Neberl, B. (eds.) *Exploring Artificial Intelligence in the New Millenium*. Morgan Kaufmann, San Francisco (2002)
7. Wang, C., Thorpe, C.: Simultaneous localization and mapping with detection and tracking of moving objects. In: *IEEE International Conference on Robotics and Automation*, pp. 2918–2924 (2002)
8. Singh, K., Fujimura, K.: Map Making by Cooperative Mobile Robots. In: *Proc. of IEEE International Conference on Robotics and Automation*, pp. 254–259 (1993)
9. Tomono, M., Yuta, S.: Mobile Robot Localization based on an Inaccurate Map. In: *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 399–405 (2001)
10. Thrun, S., Fox, D., Burgard, W., Dellaert, F.: Robust Monte Carlo localization for mobile robots. *Artificial Intelligence* 128, 99–141 (2001)
11. Brooks, R.A.: Planning Collision Free Motions for Pick and Place Operation. In: *Robotics Research*, pp. 5–38. MIT Press, Cambridge (1983)
12. Latombe, H.-L.: *Robot Motion Planning*. Kluwer Academic Publishers, Dordrecht (1991)
13. Fu, K.S., Gonzalez, R.C., Lee, C.S.G.: *Robotics*. McGraw-Hill Book Company, New York (1987)
14. Thrun, S.: Robotic mapping: A survey. In: Lakemeyer, G., Neberl, B. (eds.) *Exploring Artificial Intelligence in the New Millenium*. Morgan Kaufmann, San Francisco (2002)
15. Wang, C., Thorpe, C.: Simultaneous localization and mapping with detection and tracking of moving objects. In: *IEEE International Conference on Robotics and Automation*, pp. 2918–2924 (2002)
16. Singh, K., Fujimura, K.: Map Making by Cooperative Mobile Robots. In: *Proc. of IEEE International Conference on Robotics and Automation*, pp. 254–259 (1993)
17. Tomono, M., Yuta, S.: Object-based Localization and Mapping using Loop Constraints and Geometric Prior Knowledge. In: *Proc. of International Conference on Robotics and Automation*, pp. 862–867 (2003)
18. Brady, M., Paul, R.: *Robotics Research, The First International Symposium*. The MIT Press, Massachusetts (1984)

19. Kubota, N., Neya, H., Taniguchi, K.: Sensory Network and Evolutionary Programming for a Mobile Robot. In: Proceedings of the 4th Asia-Pacific Conference on Simulated Evolution And Learning (SEAL 2002), pp. 119–123 (2002) (CD-ROM)
20. Arulampalam, M.S., Maskell, S., Gordon, N., Clapp, T.: A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking. *IEEE Trans. Signal Processing* 50(2), 174–188 (2002)
21. Kubota, N., Yuki, K., Baba, N.: Integration of Intelligent Technologies for Simultaneous Localization and Mapping. In: The Society of Instrument and Control Engineers, SICE (2009)
22. Kohonen, T.: *Self-Organizing Maps*, 3rd edn. Springer, Heidelberg (2001)
23. Fritzke, B.: Growing Cell Structures - A Self-Organising Network for Unsupervised and Supervised Learning. *Neural Networks* 7(9), 1441–1460 (1994)
24. Hodge, V.J., Austin, J.: Hierarchical Growing Cell Structures: TreeGCS. *IEEE Trans. Knowledge and Data Engineering* 13(2), 207–218 (2001)
25. Hastie, T., Tibshirani, R., Friedman, J.: *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, New York (2001)
26. Fritzke, B.: A growing neural gas network learns topologies. In: Tesauro, G., Touretzky, D.S., Leen, T.K. (eds.) *Advances in Neural Information Processing Systems*, vol. 7, pp. 625–632. MIT Press, Cambridge (1995)
27. Fukuda, T., Kubota, N., Arakawa, T.: GA Algorithms in Intelligent Robots. In: *Fuzzy Evolutionary Computation*, pp. 81–105. Kluwer Academic Publishers, Dordrecht (1997)
28. Kubota, N., Neya, H., Taniguchi, K.: Sensory Network and Evolutionary Programming for a Mobile Robot. In: Proceedings of the 4th Asia-Pacific Conference on Simulated Evolution And Learning (SEAL 2002), pp. 119–123 (2002) (CD-ROM)
29. Fogel, D.B.: *Evolutionary Computation*. IEEE Press, Los Alamitos (1995)
30. Goldberg, D.E.: *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison Wesley, Reading (1989)
31. Syswerda, G.: A Study of Reproduction in Generational and Steady-State Genetic Algorithms. In: *Foundations of Genetic Algorithms*, pp. 94–101. Morgan Kaufmann, San Francisco (1991)
32. Hafnel, D., Burgard, W., Fox, D., Fishkin, K., Philipose, M.: Mapping and Localization with RFID Technology. In: *Proc. of the IEEE International Conference on Robotics and Automation, ICRA* (2004)
33. Bertsekas, D.P., Tsitsiklis, J.N.: *Neuro-Dynamic programming*. Athena Scientific, Belmont (1996)

Incremental Model Selection and Ensemble Prediction under Virtual Concept Drifting Environments

Koichihiro Yamauchi

Department of Information Science Chubu University
Matsumoto 1200 Kasugai Aichi, Japan
yamauchi@cs.chubu.ac.jp

Abstract. Model selection for machine learning systems is one of the most important issues to be addressed for obtaining greater generalization capabilities. This paper proposes a strategy to achieve model selection incrementally under virtual concept drifting environments, where the distribution of learning samples varies over time. To carry out incremental model selection, the system generally uses all the learning samples that have been observed until now. Under virtual concept drifting environments, however, the distribution of the observed samples is considerably different from that under real concept drifting environments so that model selection is usually unsuccessful. To overcome this problem, the author had earlier proposed the weighted objective function and model-selection criterion based on the predictive input density of the learning samples. Although the previous method described in the author's previous study shows good performances to some datasets, it occasionally fails to yield appropriate learning results because of the failure in the prediction of the actual input density. To overcome this drawback, the method proposed in this paper improves on the previously described method to yield the desired outputs using an ensemble of the constructed radial basis function neural networks (RBFNNs). Experimental results indicate that the improved method yields a stable performance.

1 Introduction

Let the learning samples be (\mathbf{x}_b, y_b) ($b = 1, 2, \dots$), whose joint probability distribution is $P(\mathbf{x}, y) = P(y|\mathbf{x})P(\mathbf{x})$. To achieve successful learning of the relation between \mathbf{x} and y : $P(y|\mathbf{x})$ using a model-based learning machine, the system generally uses all the observed samples. Although the empirical input density $P(\mathbf{x})$ approximates the actual input density with an increase in the number of learning samples, it generally differs widely from the the actual value in the early steps of the learning. Moreover, the center of $P(\mathbf{x})$ is usually nonstationary. Such changing environments are usually called "virtual concept drifting environments." Because of the underlying principle of such environments, the learning process can not yield the best model.

To overcome this problem, the author had earlier proposed a model-selection criterion on the predictive distribution of the learning samples [1] [2]. This method is an extended version of the learning strategies under *covariate shift* (e.g. [3] [4]). Under covariate shift, the learning input density $P(\mathbf{x})$ is not equivalent to the density of the test samples. In such environments, learning machines need to adjust their parameters to minimize the following weighted error function so as to acquire greater generalization capabilities.

$$\hat{E} = \sum_{i=1}^N (F(\mathbf{x}_i) - f_{\theta}(\mathbf{x}_i))^2 W(\mathbf{x}_i), \quad (1)$$

where $W(\mathbf{x})$ is the weight used for each sample and $W(\mathbf{x}) \equiv (q(\mathbf{x})/P(\mathbf{x}))^{\lambda}$, where $q(\mathbf{x})$ denotes the density of \mathbf{x} for the test samples and $0 < \lambda \leq 1$ denotes the flattening parameter. Here, $f_{\theta}(\mathbf{x})$ denotes the output of the learning machine and $F(\mathbf{x})$ denotes the target output. In incremental learning, $q(\mathbf{x})$ corresponds to the input density for all the learning samples; this includes not only the new samples introduced in subsequent phases but also the learning samples of the earlier learning phases.

Although the method proposed in the previous study shows a good performance for some datasets, it occasionally fails to yield appropriate learning results because of the failure in the prediction of the actual input density. To overcome this drawback, the method proposed in this paper improves on the previously described method to yield better learning results using an ensemble of several learning results.

The next section describes the incremental learning scheme assumed in this study. Section 3 presents a model of virtual concept drifting environments. Section 4-5 describe the incremental learning and model-selection methods used in this study. Section 6 explains the calculation of the predicted output of the system. Section 7 presents the results of synthetic and benchmark test datasets, and Section 8 provides the conclusion.

2 Learning Scheme

Let us consider the simplified incremental learning scheme shown in Fig. 1; it has a fundamental incremental learning architecture with a re-learning (rehearsal) process similar to that proposed previously [5, 6].

This system alternates between two phases, i.e., recording and rehearsal. During the recording phase, the learning system obtains a new chunk of the several new learning samples and stores these samples in a buffer having a small capacity. After the recording phase, the rehearsal phase begins. During the rehearsal phase, all the samples in the buffer and the samples generated by the previous neural network, i.e., the pseudo-old samples, which was built in the previous learning phase, are introduced to the current neural network. The initial parameters of the current neural network are obtained from the previous neural network. Note that the current neural network rehearses not only the new

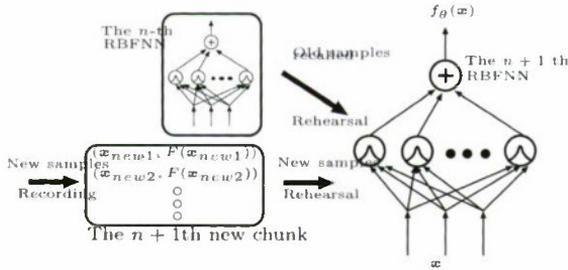


Fig. 1. Incremental learning scheme

novel learning samples but also the pseudo-old samples to prevent “catastrophic forgetting.”

The neural network used was a radial basis function neural network (RBFNN). Let $f_{\theta}(\mathbf{x})$ be the output value of the RBFNN. $f_{\theta}(\mathbf{x})$ is given by

$$f_{\theta}(\mathbf{x}) = \sum_{j=1}^M w_j \exp\left(-\frac{\|\mathbf{x} - \mathbf{u}_j\|^2}{2v_j^2}\right), \tag{2}$$

where M denotes the number of hidden units. \mathbf{u}_j and v_j^2 denote the center and variance of the j -th hidden unit, respectively. The aim of the learning system is to minimize the following evaluation function:

$$E = \int (F(\mathbf{x}) - f_{\theta}(\mathbf{x}))^2 q(\mathbf{x}) d\mathbf{x}, \tag{3}$$

where $F(\mathbf{x})$ denotes the target output and $q(\mathbf{x})$, the actual input density required to obtain the ideal learning result. Alternatively, if the actual input density is varied, $q(\mathbf{x})$ should be averaged over time. Note that $q(\mathbf{x})$ is not equivalent to empirical input density $P(\mathbf{x})$.

3 Modeling of Virtual Concept Drifting Environments

In order to devise a learning method to minimize the weighted error function given by Eq(1), we need to derive the (average) actual input density $q(\mathbf{x})$ and empirical input density $P(\mathbf{x})$. It is essential to predict $q(\mathbf{x})$ beforehand using the given learning samples.

3.1 Prediction of $q(\mathbf{x})$

The following predicted distribution of \mathbf{x} from N number of learning samples, which have been presented up till now, is used in this study. Let $\hat{q}(\mathbf{x})$ be the predicted $q(\mathbf{x})$.

$$\hat{q}(\mathbf{x}) = \int P(\mathbf{x}|S)P(S|\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N)dS, \tag{4}$$

where S denotes the parameter vector that represents the input density function. In the simplest case, $q(\mathbf{x})$ would be a Gaussian probability distribution. In such cases, according to Bayes theorem, $\hat{q}(\mathbf{x})$ should be approximated by a *Student's-t* distribution of $(N - 1)$ -degrees-of-freedom. Therefore,

$$\hat{q}(\mathbf{x}) = \frac{\Gamma[(N - 1 + p)/2]}{((N - 1)\pi)^{p/2} \Gamma[(N - 1)/2] |\Sigma|^{1/2}} \left[1 + \frac{(\mathbf{x} - \mathbf{u})^T \Sigma^{-1} (\mathbf{x} - \mathbf{u})}{N - 1} \right]^{-(N-1+p)/2} \tag{5}$$

where p is the number of input dimension, $\mathbf{u} = E[\mathbf{x}]$, $\Gamma[\cdot]$ denote gamma function and Σ denote the scale matrix. The scale matrix is described by $\Sigma = ((n - 3)/n)C$, $C = E[(\mathbf{x} - \mathbf{u})(\mathbf{x} - \mathbf{u})^T]$ [7]. Note that *Student's-t* distribution converges to actual Gaussian distribution $q(\mathbf{x})$ while increasing the number of presented learning samples.

In many cases, however, the center of the actual input density is usually moving overtime so the averaged density is difficult to be approximated by using a single *Student's-t* distribution. To overcome this difficulty, we extended Eq.(5) to supporting more complex input distributions.

Let us imagine the sensory inputs of a robot. In such an actual environment, each sample is highly related to the current state. Therefore, the learner observes many similar samples within a short interval of time where the state of the robot remains almost the same. If the robot moves to another location, however, its state is changed and the input distribution is also changed. Similar situations to these frequently appear in actual incremental-learning environments.

Let us denote the current state S_i ($i = 1, 2, \dots$). Each state, S_i , is represented by the corresponding position of input space. We assume that the state will change during certain periods but will return to the same state after a prolonged period(e.g. Fig 2).

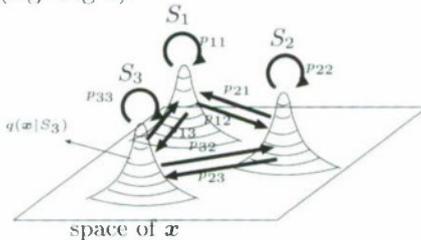


Fig. 2. State transition between input distributions. Each state, S_i , is represented as the corresponding position of input space. p_{ij} denotes the state transition between the i - and j -th states, and $q(\mathbf{x}|S_i)$ is the Gaussian distribution.

Therefore, this is an ergodic Markov process and means that the probability for each S_i converges to a certain value $p(S_i)$ which does not depend on the initial state¹. From this, $\hat{q}(\mathbf{x})$ can be approximated as

$$\hat{q}(\mathbf{x}) \simeq \sum_i q(\mathbf{x}|S_i)p(S_i). \tag{6}$$

Therefore, $\hat{q}(\mathbf{x})$ can be represented as a mixture of distributions. Similarly, $\hat{P}(\mathbf{x})$ is given by

$$\hat{P}(\mathbf{x}) \simeq \sum_i P(\mathbf{x}|S_i)p(S_i). \text{ Therefore, } \frac{\hat{q}(\mathbf{x})}{\hat{P}(\mathbf{x})} = \frac{\sum_i q(\mathbf{x}|S_i)p(S_i)}{\sum_i P(\mathbf{x}|S_i)p(S_i)}. \tag{7}$$

¹ We assume that the time interval for state transition is considerably longer than that for presenting each sample.

where $P(\mathbf{x}|S_i)$ and $q(\mathbf{x}|S_i)$ represent the Gaussian distribution and *Student's-t* distributions, respectively; the center of $q(\mathbf{x}|S_i)$ coincides with that of $P(\mathbf{x}|S_i)$. The calculation of Eq(7) requires the precise addition of the coefficients of the *Student's-t* distributions; this can be approximated without utilizing coefficients under the assumption that the effect of tails of the distributions is low.

$$\frac{\hat{q}(\mathbf{x})}{P(\mathbf{x})} \simeq \frac{q(\mathbf{x}|S_j)p(S_j)}{P(\mathbf{x}|S_j)p(S_j)} = \frac{q(\mathbf{x}|S_j)}{P(\mathbf{x}|S_j)}, \text{ where } j = \arg \max_i P(\mathbf{x}|S_i). \quad (8)$$

$p(\mathbf{x}|S_i)$ is also a Gaussian distribution $\mathcal{N}(\mathbf{u}_i, \Sigma_i)$. The center \mathbf{u}_i and variance-covariance matrix Σ_i are determined by using an incremental Expectation and Maximization (EM)-algorithm, which is an improved version of Ref[8]. The incremental EM-algorithm is not the same as the online EM algorithm because the method needs to work even if the distribution of inputs is not i.i.d. samples. The detailed algorithm is explained in the next section.

In other words, a Gaussian mixture distribution is constructed with the EM algorithm but only the resulting \mathbf{u}_i and Σ_i are used in the following method. In this case, the appropriate number of Gaussians should also be determined by using an information-criterion such as AIC[9]. Therefore, the Gaussian mixture distribution having the smallest AIC value is the appropriate data-model.

Then, the estimate is applied to all Gaussian distributions in the resulting mixture model. Therefore, if the likelihood of \mathbf{x} for the i -th Gaussian is the maximum of all likelihoods, the corresponding $W(\mathbf{x})$ is

$$W(\mathbf{x}) = \left\{ \left(\frac{2}{N_i - 1} \right)^{p/2} \frac{\Gamma[(N_i + p - 1)/2]}{\Gamma[(N_i - 1)/2]} \frac{\left[1 + \frac{(\mathbf{x} - \mathbf{u}_i)^T \Sigma_i^{-1} (\mathbf{x} - \mathbf{u}_i)}{N_i - 1} \right]^{-(N_i + p - 1)/2}}{\exp \left(-\frac{1}{2} (\mathbf{x} - \mathbf{u}_i)^T \Sigma_i^{-1} (\mathbf{x} - \mathbf{u}_i) \right)} \right\}^\lambda, \quad (9)$$

where $i = \arg \max_j \frac{1}{(2\pi)^{p/2} |C_j|^{1/2}} \exp \left(-\frac{(\mathbf{x} - \mathbf{u}_j)^T C_j^{-1} (\mathbf{x} - \mathbf{u}_j)}{2} \right)$. (10)

In the above two equations, N_i , Σ_i , and \mathbf{u}_i correspond to the degree of freedom, the scale matrix, and mean input vector of the i -th Gaussian distribution. The scale matrix is described by $\Sigma_i = ((n - 3)/n)C_j$, $C_j = E[(\mathbf{x} - \mathbf{u}_j)(\mathbf{x} - \mathbf{u}_j)^T]$ [7]. The degree of freedom was set to $N_i = \sum_n \left\{ P(\mathbf{x}_n|S_i) / \sum_j P(\mathbf{x}_n|S_j) \right\}$ for simplicity.

Note that this method approximates $q(\mathbf{x})$ using given samples. Consequently, if the number of given samples is too small, it is hard to accurately approximate $q(\mathbf{x})$. The method only approximates $q(\mathbf{x})$ where \mathbf{x} is near to one of the learned samples. Although it has the above drawback, the method is adequate for learning RBFNN. This is because the RBFNN eventually yields proper outputs only for \mathbf{x} 's near to the learned samples.

4 Incremental EM-Algorithm

To predict $q(\mathbf{x})$, the system needs to execute the EM-algorithm to construct a Gaussian mixture model for $P(\mathbf{x})$. The original EM-algorithm, however, needs to store whole learning samples in advance to execute the algorithm. To save the storage space and computational power, an online version of the EM algorithm would be suitable for this system. Unfortunately, current online EM algorithms (e.g. [10]), are designed to forget past learning results to adjust to the current input distribution. This property is not suitable for handling the virtual-concept drifting environments.

In the virtual-concept drifting environments, the model parameters should be adjusted not only to the new learning samples but also to the old learning samples. Therefore, the system needs the old learning samples as well as the new ones to reconfigure the model parameters. To overcome the problem, the system uses pseudo-samples generated by the RBFNN, which was constructed in the previous rehearsal phase, instead of using the real old learning samples. The pseudo-sample generation algorithm is to be explained in section 5.2.

The number of the Gaussians should also be an optimal number to get greater generalization capability. To search such data-model quickly, the EM-algorithm and the AIC estimation are applied while adding the number of Gaussians one by one to the preceding best model until the current AIC value becomes larger than that of the previous one. Then, the previous model is derived as the ultimate Gaussian mixture model.

5 Incremental Learning and Model Selection for RBFNN

The RBFNN learns samples stored in the buffer and pseudo-samples generated from the previous RBFNN in each rehearsal phase. As discussed in Section 1, the RBFNN has to minimize the weighted error function i.e., Eq.(1).

In this study, a modified version of the quick RBFNN learning method proposed by Moody and Darken 1989 [11] was used because it ensured that the output connection strengths were always optimal values, which minimized the error function, under a corresponding setting for hidden units. Moreover, it could also support various numbers of hidden units, which were fewer than those of the learning samples. The appropriate number of hidden units was selected using an information criterion, IC_w , described in section 5.4.

5.1 Learning of First Chunk

In the modified RBFNN method, the centers and variances of the RBF hidden units are determined using a weighted fuzzy k -means algorithm, whereas the connection weights between the RBF hidden units and the output unit are determined by a weighted least squares (WLS) method.

The weighted fuzzy k -means algorithm, which is an extended version of the fuzzy k -means algorithm [12], updates cluster center \mathbf{u}_j not only according to the cluster centers obtained in the previous step but also according to the weight

of each sample, as given in the equation below. Note that the cluster center is the center of each hidden unit of the RBFNN.

$$\mathbf{u}_j^{(n+1)} := \sum_{b=1}^N \frac{W(\mathbf{x}_b)\mathbf{x}_b \exp(-\|\mathbf{x}_b - \mathbf{u}_j^{(n)}\|^2/(2c^2))}{\hat{c}_w \sum_{j'} \exp(-\|\mathbf{x}_b - \mathbf{u}_{j'}^{(n)}\|^2/(2c^2))}, \tag{11}$$

where $\hat{c}_w = \sum_{b=1}^N W(\mathbf{x}_b)$ and c is the standard deviation. For simplicity, the initial centers, $\mathbf{u}_j^{(0)}$, are set to the first k samples, i.e., \mathbf{x}_j , in the buffer B . After converging the weighted fuzzy k -means algorithm, the variance of each hidden unit is set to

$$\sigma_j^2 = \kappa \min_{j' \neq j} \|\mathbf{u}_j - \mathbf{u}_{j'}\|^2, \tag{12}$$

where $\kappa (> 0)$ denotes the overlapping factor [13].

The WLS derives the optimized output connection vector $\mathbf{w}_{ML} = (w_1, w_2, \dots, w_M)^T$, where w_i denotes the connection weight between the i -th hidden unit and the output unit that analytically minimizes Eq(1). Therefore,

$$\mathbf{w}_{ML} = (\Phi_0 \mathbf{W}_0^T \Phi_0)^{-1} \Phi_0^T \mathbf{W}_0 \mathbf{F}_0, \tag{13}$$

where \mathbf{F}_0 denotes target output vector of the first chunk ($\mathbf{F}_0 = (F(\mathbf{x}_1), F(\mathbf{x}_2), \dots, F(\mathbf{x}_{N_0}))^T$) and \mathbf{W}_0 is a diagonal matrix, whose diagonal elements are given by $W_{0\ bb} = W(\mathbf{x}_b)$ ($b = 1, 2, \dots, N_0$). Φ_0 is the design matrix of the first chunk, whose elements are given by $\Phi_{0bj} = \exp(-\|\mathbf{x}_b - \mathbf{u}_j\|^2/(2\sigma_j^2))$. Using the modified RBFNN method, the learning system ensures that the output connections will always have optimal weights, so that we can accurately estimate the effect of the weighted error function given by Eq (1).

5.2 Pseudo Sample Generation

The WLS method essentially requires all the samples to construct a design matrix Φ_i , which is used in Eq(13). However, the recording all the samples consumes huge storage space in the later steps of learning. To overcome this problem, we should regenerate them as pseudo-samples. One method of generating a pseudo-sample is to use the center of the hidden unit and the corresponding output of the RBFNN as proposed in [5] [14]. However, two problems are encountered in applying such methods for this learning system:

1. This model reduces the number of hidden units through model selection therefore the number of pseudo samples will also be reduced. A small number of pseudo samples usually yields poor learning results.
2. The pseudo-sample distribution generated by the former models is not equivalent to the original sample distribution. This also degrades the system performance.

To overcome these problems, the system stores the RBFNN parameters, that were determined in the previous rehearsal phase, and the key information of the learned samples. The key information for the p -th learning sample is (j_p, F_p) ,

where $j_p = \arg \max_{\alpha} \phi_{\alpha}(\mathbf{x}_p)$ and $F_p = f_{\theta_{n-1}}(\mathbf{x}_p)$, where $f_{\theta_{n-1}}(\mathbf{x}_p)$ denotes the previous RBFNN output for \mathbf{x}_p . Note that the system only needs to save a single two-dimensional data (j_p, F_p) for one learning sample. Therefore, lesser storage space is used for saving this information than for saving the real learning samples if the number of dimension for \mathbf{x}_p is larger than two. Using RBFNN parameters of the previous rehearsal phase, the system can approximately re-generate the p -th sample by minimizing the difference between the recorded output and the RBF output values as follows:

$$\hat{\mathbf{x}}_p := \hat{\mathbf{x}}_p - \eta \frac{\partial \{F_p - f_{\theta_{n-1}}(\hat{\mathbf{x}}_p)\}^2}{\partial \hat{\mathbf{x}}_p}, \tag{14}$$

where η denotes the varying speed of $\hat{\mathbf{x}}_p$. This is known as the gradient descent method, where the convergence speed depends on the value of η . The initial vector of $\hat{\mathbf{x}}_p$ is set to $\mathbf{u}_{j_p} + \epsilon$, where j_p is the key information for the p -th learning sample and ϵ is a small random vector. The method is repeated until convergence. Thereafter, $\hat{\mathbf{x}}_p$ can be used as the p -th pseudo-learning sample.

5.3 Incremental Weighted Least Squares

If the system receives the n -th new chunk, it creates a clone of the provisional best RBFNN, which was constructed in the previous rehearsal phase, as the new learner. Then, the new learner learns not only the new samples but also the old samples, which are recalled from the provisional best RBFNN. At first, m' new hidden units are appended to the learner. Then, the system configures the hidden unit centers for the new hidden units as well as the old hidden units. This process is achieved by applying the weighted fuzzy k -means method (see 5.1) to the above hidden units using both the new learning samples and the pseudo learning samples generated by the procedure described in 5.2.

After the configuration of the hidden unit centers, Φ_n is generated as follows.

$$\Phi_n = \begin{bmatrix} \phi_1(\hat{\mathbf{x}}_1) & \cdots & \phi_{m_n}(\hat{\mathbf{x}}_1) \\ \phi_1(\hat{\mathbf{x}}_2) & \cdots & \phi_{m_n}(\hat{\mathbf{x}}_2) \\ \vdots & \cdots & \vdots \\ \phi_1(\hat{\mathbf{x}}_{N_{n-1}}) & \cdots & \phi_{m_n}(\hat{\mathbf{x}}_{N_{n-1}}) \\ \phi_1(\mathbf{x}_1) & \cdots & \phi_{m_n}(\mathbf{x}_1) \\ \vdots & \cdots & \vdots \\ \phi_1(\mathbf{x}_{N_c}) & \cdots & \phi_{m_n}(\mathbf{x}_{N_c}) \end{bmatrix} \tag{15}$$

where $\hat{\mathbf{x}}_p$ and \mathbf{x}_n denote the p -th pseudo sample vector generated by $RBFNN(n-1)$ and the n -th new samples in the new chunk. N_{n-1} denotes the total number of learned samples until the previous rehearsal phase, and N_c is the number of the new learning

samples in the current new chunk. Note that $N_n = N_{n-1} + N_c$.

After the generation of Φ_n , the weight connections between the hidden units and the output unit are derived as follows.

$$\mathbf{w}_{ML} = (\Phi_n \mathbf{W}_n^T \Phi_n)^{-1} \Phi_n^T \mathbf{W}_n \mathbf{F}_n, \tag{16}$$

where \mathbf{W}_n and \mathbf{F}_n are the weights and desired outputs of all samples, respectively. They are created by expanding the size of \mathbf{W}_{n-1} and \mathbf{F}_{n-1} and setting the corresponding weights and outputs for the new samples².

Note that m' , the number of the added hidden units is also optimized by using the information criterion described in the next section. The transition of the number of hidden units is similar to that of the incremental EM.

5.4 Determination of λ and Number of Hidden Units

The flattening parameter λ and number of hidden units M must be determined properly to attain greater generalization capabilities. In this study, the information criterion IC_w , which was proposed by Shimodaira (2000)[3], was used for determining these λ and M . IC_w is used to estimate the performance of a learning machine using the weighted error function under covariate shift. Therefore, the system searches (λ_*, M_*) to ascertain which IC_w value is the minimum. In the experiment, we prepared several sets of λ and m' . They were applied to construct the new RBFNN, and the resulting RBFNNs were estimated using IC_w .

6 Ensemble Prediction of Output

The quality of the resultant RBFNN is highly affected by the accuracy of the predictive distribution, which in turn, depends on the variation in the given learning samples. As a result, the resultant RBFNN performance is occasionally lower than that of the original RBFNN, which does not employ the weighted error function given by Eq.(1) [1] [2].

To overcome this drawback, the output of the proposed system is considered to be an ensemble of the outputs of the following two RBFNN: (a) the RBFNN, that learned the samples using the proposed method described in Section 4-5, and (b) the original RBFNN, which learned the samples by the ordinary least squares(OLS) method. Therefore, the ultimate output $\hat{f}(\mathbf{x})$ is given by

$$\hat{f}(\mathbf{x}) = w_{WLS} f_{\theta_n^{WLS}}(\mathbf{x}) + w_{OLS} f_{\theta_n^{OLS}}(\mathbf{x}), \quad (17)$$

Note that $f_{\theta_n^{OLS}}(\mathbf{x})$ executes the incremental learning procedure in the same way as $f_{\theta_n^{WLS}}(\mathbf{x})$ except that it uses the normal objective function. In Eq.(17), w_{WLS} and w_{OLS} denote the weights for the two RBFNNs. Their values are 0.5 immediately after completion of the learning phase; however they are sequentially modified according to the square of the errors of the new samples introduced in the succeeding recording phase.

$$w_{WLS} = \frac{\exp\left(-\frac{\hat{e}_{WLS}}{\sigma}\right)}{\exp\left(-\frac{\hat{e}_{WLS}}{\sigma}\right) + \exp\left(-\frac{\hat{e}_{OLS}}{\sigma}\right)}, \quad w_{OLS} = \frac{\exp\left(-\frac{\hat{e}_{OLS}}{\sigma}\right)}{\exp\left(-\frac{\hat{e}_{WLS}}{\sigma}\right) + \exp\left(-\frac{\hat{e}_{OLS}}{\sigma}\right)}, \quad (18)$$

² The weights for the old samples are reused in the subsequent learning phase for simplicity.

where \hat{e}_{WLS} and \hat{e}_{OLS} are the mean square errors of the two RBFNNs. They vary according to the square of the error for each new sample, e.g.,

$$\hat{e}_{WLS} := \hat{e}_{WLS} + \frac{1}{N} \left[\left\{ F[\mathbf{x}_N] - f_{\theta_{WLS}}(\mathbf{x}_N) \right\}^2 - \hat{e}_{WLS} \right], \quad (19)$$

where N denotes the index of the new sample. Note that the RBFNN having greater weight is the provisional best RBFNN for the succeeding rehearsal phase.

7 Experiments

The system was tested using one synthetic dataset and two benchmark test datasets. For convenience, the RBFNN, that uses the weighted error function, was denoted as ‘‘WRBFNN’’ in these experiments. The performance of the WRBFNN was compared with that of the original RBFNN, which does not use the weighted error function. The original RBFNN is denoted as ‘‘org-RBFNN’’ hereinafter. Org-RBFNN is equivalent to the fundamental architecture of nearly all the former incremental learning systems that use RBFNN or perceptron [6, 14–16]. Note that org-RBFNN is the same as WRBFNN for $\lambda = 0$.

7.1 Illustrative Example in One-Dimensional Synthetic Dataset[2]

The following simple dataset was used to accurately evaluate the system behavior. $(x, y) = (x, 1.5)$ where $x \sim \mathcal{N}(-20, 2)$ or $\mathcal{N}(20, 2)$

Note that $F(x) = y = 1.5$. There were 101 learning samples. One isolated point $(x, y) = (10, 1.5)$ was manually added to clearly demonstrate the effects of the weighting function, given by Eq(9). The system learned two chunks of the data sequentially. The first chunk consisted of 50 samples generated from $\mathcal{N}(-20, 2)$. The second one consisted of the isolated point $(x, y) = (10, 1.5)$ and 50 samples generated from $\mathcal{N}(20, 2)$. The overlap factor, κ , was set to 2 for both WRBFNN and org-RBFNN. We compared the performances of WRBFNN and org-RBFNN. After the second rehearsal phase, the proposed system yielded the WRBFNN having four hidden units that learned the samples for $\lambda = 1$.

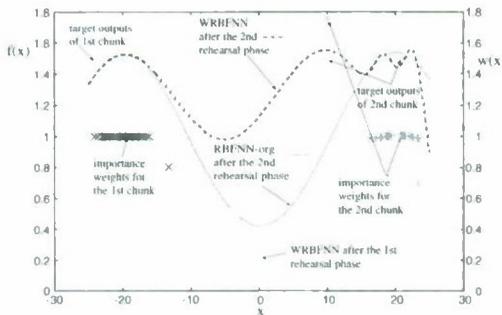


Fig. 3. Output curves for WRBFNN and org-RBFNN

Fig. 3 shows the output curves for WRBFNN and org-RBFNN. We can see that the weight for each learning sample is ≤ 1 when x is close to the edges. In particular, the weight for the isolated point is considerably greater than those for the other samples. This means that if the learning samples appear infrequently, the corresponding weight increases. However, org-RBFNN does not learn such samples well due to their low frequency of appearance. By using the

proposed approach, WRBFNN can learn such samples better than org-RBFNN on account of the increase in the corresponding weights. Consequently, it can be observed from Fig. 3 that output curve for WRBFNN fits the isolated points. Note that, this test did not use the ensemble prediction method for evaluating the effect of importance weight clearly. Therefore, the proposed system had interpreted that the isolated sample is the prelude to the change of input distribution and similar samples to the isolated one will be introduced in the near future.

7.2 System Behavior with Benchmark Dataset

To verify the validity of the proposed method, the performance of the system was examined with regard to the benchmark datasets for regression, i.e., Auto mpg, CPU performances and Servo of the University of California, Irvine (UCI) machine learning repository. The performance of the proposed system was compared with that of the previous system proposed in [2].

The parameters used in both the systems were $\sigma = 0.1$ (Eq(18)) and $\kappa = 2.5$ (Eq(12)). In this experiment, the variance covariance matrix, Σ_j , for the Gaussian mixture model was used as a diagonal matrix for simplicity.

In the both the datasets, 50 learning samples were randomly selected from each dataset for each chunk. The two systems repeated the rehearsal phase three times. All the samples were used as test samples. This test was repeated 50 times for different learning datasets. To prevent the learning process from becoming unstable, the weight for each sample was restricted to ≤ 10 .

In the case of the previous system, each result was plotted as a two dimensional point, $(x, y) = (MSE_{WRBFNN}, MSE_{org-RBFNN})$, where MSE_* denotes the mean square error calculated by using all the samples in the dataset. Note that if WRBFNN outperforms org-RBFNN, the points are located above the line $y = x$. In the case of the proposed new one, the result was plotted as $(x, y) = (MSE_{ultimate}, MSE_{org-RBFNN})$, where $MSE_{ultimate}$ denotes the mean square error of the combined outputs defined in Eq.(17) after the introduction of new samples in the succeeding phase.

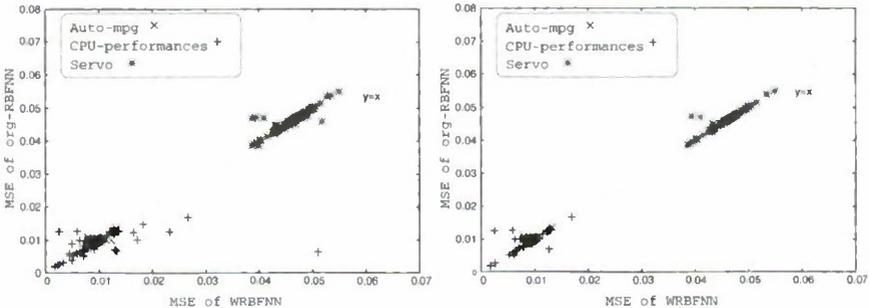


Fig. 4. System performances for Auto mpg, CPU performances and Servo after first and second rehearsal phases. Left: Previous system. Right: Proposed system.

Fig. 4 shows the responses of the two systems for the two datasets after the first and second rehearsal phases. From this figure, we can see that the performance of the previous system is usually lower than that of org-RBFNN in case of the CPU performance and Servo datasets. On the other hand, the proposed system achieves good performance for both the datasets. This means that the proposed system adaptively chooses the better RBFNN according to the mean square error of the new samples.

8 Conclusion

In this study, we attempted to develop an incremental learning system based on the predictive distribution of virtual concept drifting environments. The new approach was able to predict the input density of the new learning samples, that were introduced in later incremental learning steps. This made the learning system undergo proactive learning according to the predicted input density. Therefore, the new incremental learning scheme reinforces the learning effect using novel isolated learning samples.

The proposed system is an improved version of previous systems [1] [2]. The main difference between the previous systems and the proposed one is that the latter incorporates an ensemble prediction mechanism to obtain a stabler recognition ability. Experimental results demonstrated that the likelihood of failure of learning using this system is reduced. The system, however, needs to adjust the connection weights for the ensemble using the new samples introduced in succeeding recording phases.

References

1. Yamauchi, K.: Optimal incremental learning under covariate shift. *Memetic Computing* 1(4), 271–279 (2009)
2. Yamauchi, K.: Incremental learning and model selection under virtual concept drifting environments. To appear in the 2010 IEEE World Congress on Computational Intelligence (IEEE WCCI 2010) (2010)
3. Hidetoshi, S.: Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference* 90(2), 227–244 (2000)
4. Sugiyama, M., Nakajima, S., Kashima, H., von Bünau, P., Kawanabe, M.: Direct importance estimation with model selection and its application to covariate shift adaptation. In: *Twenty-First Annual Conference on Neural Information Processing Systems (NIPS 2007)* (December 2007)
5. Yamauchi, K., Hayami, J.: Incremental learning and model selection for radial basis function network through sleep. *IEICE Transactions on Information and Systems* E90-D(4), 722–735 (2007)
6. French, R.M.: Pseudo-recurrent connectionist networks: An approach to the “sensitivity stability” dilemma. *Connection Science* 9(4), 353–379 (1997)
7. López-Rubio, E.: Multivariate student-t self-organizing maps. *Neural Networks* 22, 1432–1447 (2009)

8. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society B39*(1), 1–38 (1977)
9. Akaike, H.: A new look at the statistical model identification. *IEEE Transactions on Automatic Control AC-19*(6), 716–723 (1974)
10. Sato, M., Ishii, S.: On-line EM algorithm for the normalized Gaussian network. *Neural Computation* 12, 407–432 (2000)
11. Moody, J., Darken, C.J.: Fast learning in neural networks of locally-tuned processing units. *Neural Computation* 1, 281–294 (1989)
12. Bezdek, J.C.: A convergence theorem for the fuzzy isodata clustering algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2, 1–8 (1980)
13. Platt, J.: A resource allocating network for function interpolation. *Neural Computation* 3(2), 213–225 (1991)
14. Yamauchi, K., Yamaguchi, N., Ishii, N.: Incremental learning methods with retrieving interfered patterns. *IEEE Transactions on Neural Networks* 10(6), 1351–1365 (1999)
15. Yamakawa, H., Masumoto, D., Kimoto, T., Nagata, S.: Active data selection and subsequent revision for sequential learning with neural networks. In: *World Congress of Neural Networks (WCNN 1994)*, vol. 3, pp. 661–666 (1994)
16. Ozawa, S., Toh, S.L., Abe, S., Pang, S., Kasabov, N.: Incremental learning of feature space and classifier for face recognition. *Neural Networks* 18, 575–584 (2005)

Multi-dimensional Data Inspection for Supervised Classification with Eigen Transformation Classification Trees*

Steven De Bruyne¹ and Frank Plastria²

¹ Vrije Universiteit Brussel
Steven.De.Bruyne@vub.ac.be

² Vrije Universiteit Brussel
Frank.Plastria@vub.ac.be

Abstract. Data visualisation can be a great support to the data mining process. We introduce a data structure that allows browsing through the data giving a complete but very manageable overview over the entire data set, where the data is split into subsets and displayed from interesting angles to reveal the relevant patterns for each subset.

Based on the features originating from principal separation analysis, a tree is grown. A node of the tree is associated with a feature and a subset of instances, and later on with a two-dimensional visualisation. At the node level, groups of instances of different classes that can be displayed from a more interesting angle are temporarily grouped together in subsets. For each of these subsets child nodes are created that display this part of the data from a more interesting angle, revealing new patterns. This process is continued until no further improved visualisation can be found.

After the tree has been constructed, it can be used to easily browse through the data. The nodes correspond with two-dimensional visualisations of the data, but the specific properties of the tree allow for three-dimensional animated transitions from one node to another, further clarifying the patterns in the data.

1 Introduction

Visualizing data can give a data miner already a good idea of the structure of the data. The largest problem is that only two-dimensional images are easily interpretable for the human eye. Unfortunately, data sets tend to have far higher dimensionalities, so that a single two-dimensional image does not suffice. Therefore, multiple visualisations are needed, such as e.g. a scatterplot matrix [2]. But as each combination of two attributes is visualised here, the user also gets rapidly overwhelmed by the information overflow, even if the number of attributes is low.

A solution is to show the data from the most interesting angles by using e.g. principal component analysis [7] or Fisher's linear discriminant analysis [5]. But

* Partially supported by the OZR1372 project of the Vrije Universiteit Brussel.

this has also some severe limitations. By using only the first two axes, only a part of the structure of the data is visible. This may suffice for simple binary problems, but once the data is more complex or when there are more than two classes, one two-dimensional image does not suffice. In these cases the other axes also hold important information about the less predominant patterns. Creating more images using these other axes will bring little relief as the other instances will obscure the pattern for the instances for which the patterns are relevant.

Therefore we introduce a classification tree, the eigentransformation classification tree, whose function is not to classify, but to hold views on parts of the data from the most important angles. The angles are also derived by an eigen transformation. A big difference with other techniques that view data from multiple interesting angles, such as The Grand Tour [1], is that only those subsets of the data for which the current view is relevant are displayed. The structure of the tree also allows to smoothly move from one visualisation to another, further simplifying the interpretation for the data miner [6].

2 Principal Separation Analysis

The eigen transformation we use is called principal separation analysis (PSA), which was introduced in [8], but only for two classes. We extend the principle here to multiple classes. Consider that our global dataset $X \subset \mathbb{R}^d$ is partitioned into the classes $P \in \mathcal{P}$. We are only interested in considering intraclass vectors and distances. So define the intraclass differences set $D(\mathcal{P})$, consisting of all d -vectors $p - q$ for any pair of d -vectors $p \in P$ and $q \in Q$ for $P \neq Q \in \mathcal{P}$. This may also be written as follows:

$$D(\mathcal{P}) = \{ p - q \mid (p, q) \in (X \times X) \setminus \bigcup_{P \in \mathcal{P}} (P \times P) \}$$

Using the notation of [8] we then define the reduction matrix for multiclass principal separation components as

$$R = \text{Eig}(\text{Mom}(D(\mathcal{P})))$$

We prefer PSA to principal component analysis and Fisher's linear discriminant analysis, because contrary to principal component analysis it takes the classes into account, and contrary to Fisher's linear discriminant analysis it does not run into numerical problems if there are linear dependencies.

3 Eigen Transformation Classification Trees

The PSA will yield eigen vectors that allow us to identify important patterns. The eigen vectors with the largest eigen values will reveal the most significant patterns relevant for the entire data set, while the eigen vectors with smaller eigen values will indicate patterns that are only relevant for smaller subsets.

We define an eigen transformation classification tree (ETCT) that will work in a feature space with features based on the PSA matrix, based on some preliminary work that resulted in a classifier called a 2-class eigen transformation classification tree (2C-ETCT) [3]. The tree will subsequently split the space along the features following the order indicated by the eigen values. It will check for neighbouring groups whether the pattern at the current level is the most significant to separate them, which will result in a split at that level, or if there exists a more specific pattern further down, which will result in keeping the groups together at the current moment. The tree will finally be evaluated to prune away those parts that fit the specific instances of the data but not necessarily the patterns in the data.

The ETCT can then be used to browse through the data. 2D views are generated based on the feature corresponding with the level of the node and its parent node, yielding the most interesting view for the subset. The splits can also be displayed on the same view, creating zones which the user can select to investigate the corresponding subset from a more interesting angle. The structure of the tree also makes it possible to move from one view to another seamlessly through 3D animations.

4 Creating the Tree

4.1 Data Model

For each node the level, the expected class and the splits are stored. The level indicates the corresponding feature, i.e. the feature derived from the eigenvector, at the index equal to the level, of the PSA matrix sorted by the eigenvalues. The expected class is the class to which an instance reaching this node most likely belongs. The splits are the values to which the feature of an instance corresponding with the level is compared to decide to which child node it should be sent.

Each node also has ten folds, which contain the information that is used to prune the tree. Each fold corresponds with one pair of a training set and a test set. They store the expected class and the splits based on the training set, and the number of correct and incorrect based instances of the test set based on the results on the training set.

A node also has a class transform structure, which is used to merge different classes of the classification problem together at the level of a node. This is done to merge classes temporarily together if the instances belonging to these classes can be better split deeper in the tree. A global class of the classification problem will correspond with a local class on node level. Many global classes can map to the same local class.

4.2 Algorithm

The creation of the tree starts with the computation of the PSA matrix based on the data points passed as a parameter. During the next step the instances

are transformed using the matrix, yielding instances with new features. Then the top node is created and a grow message is sent to it with all the transformed instances as a parameter. A first pruning takes place before the tree is evaluated. Then the transformed data set is used to create training and test folds based on two 5-fold crossvalidations. Each pair of training and test sets is used to send an evaluation message to the top node. The information generated during the evaluation is then used for a final pruning.

```

Create the PSA transformation matrix using all data points
transformed data set ← data set × transformation matrix
Create the top node
top.Grow(transformed data set)
top.PruneBeforeEvaluation
Create the internal training and test folds from
  the transformed data set
For each pair of training and test sets do
  top.Evaluate(fold index, training set, test set)
End for
top.PruneAfterEvaluation

```

5 Browsing the Tree

After the ETCT is built, it can be used to visually browse through the data set. A 2D view is generated based on the information of a node and the instances that reached that node after applying the splits in the nodes above. The user can move to a node below by left clicking the corresponding zone on the current 2D view. This will trigger a 3D animation that will zoom in on the selected zone, add information of the selected node and rotate the information of the previous node away, similar to the technique used in [4]. By right clicking the visualisation, the user moves up a node in a similar fashion, where information of the newly selected node is rotated in and information of the previous node rotated out, followed by a zoom out.

6 Example

Advantages of the ETCT such as (1) limiting the number of views on the data, (2) finding the more interesting angles and (3) the possibility to move from one view to another through an animation are direct consequences of the properties of the structure of the ETCT. The final advantage of the ETCT against other visualization techniques is that (4) smaller patterns are made visible as only the instances to which the pattern applies are shown. To illustrate this, we use an artificial data set with 1000 instances, 7 classes and 4 attributes. Each group is largely linearly separable from each other group along one of the attributes except for the green and cyan groups, which overlap for all attributes. The combination of groups and the attribute that separates them are the patterns we

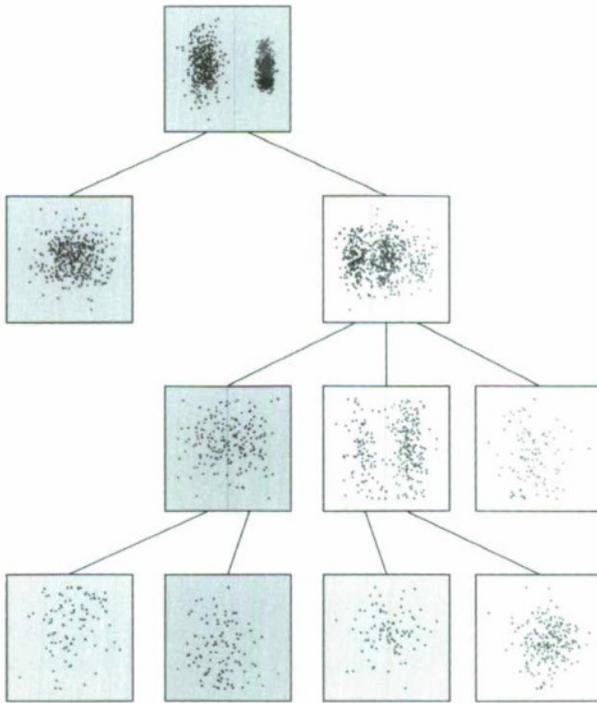
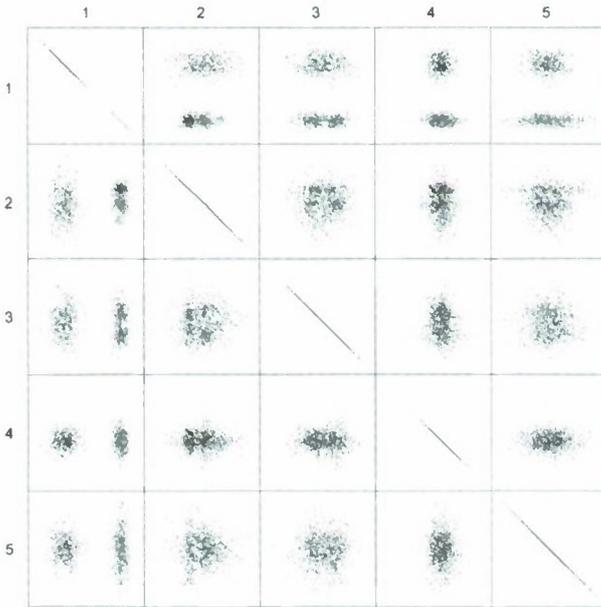


Fig. 1. Top: Scatterplot Matrix; Bottom: Eigen Transformation Classification Tree

are looking for. Moreover, some patterns are more significant than other. Forty percent of the instances belong to the red class, and can be separated from all other groups using the first attribute. The other groups are equal in size, but the second attribute allows separating the violet and indigo classes from the blue, cyan, green and yellow classes; as well as separating the yellow class from the violet, indigo, blue, cyan and green classes. The other attributes only separate two of the smaller groups. This data set also nullifies the other advantages of the ETCT as the number of attributes is limited (relevant for advantage (1)), none are redundant (relevant for advantage (2)) and each pattern is expressed uniquely by one attribute (relevant for advantage (2)), thereby only illustrating the ability to reveal smaller patterns (advantage (4)). For all these reasons is the corresponding scatterplot matrix shown in figure 1 very manageable, complete and shows the data from the most interesting angles. Therefore can no other technique that uses the full data yield better results. When we evaluate the visualisation techniques based on visibility of the patterns, we observe that only the scatterplots in the upper left reveal clear patterns while additional information is hard to discern in the scatterplots located more in the lower right.

Figure 1 also shows the ETCT of the same data set, where the nodes are represented by their respective 2D views. By removing the instances for which the orientation is not relevant, the smaller patterns are clearly visible without any redundant views. An animated depth first exploration of this tree can be found at <http://homepages.vub.ac.be/~sdebruyn/etct/etct.avi> (format: XVID), illustrating both the moving down and moving up animations. The animations link the more informative 2D views, making the data even more understandable for the user.

References

1. Asimov, D.: The Grand Tour. *SIAM Journal on Scientific and Statistical Computing* 6(1), 128–143 (1985)
2. Cleveland, W.S., McGill, M.E.: *Dynamic Graphics for Statistics. Statistics/Probability Series.* Wadsworth & Brooks/Cole, Pacific Grove (1988)
3. De Bruyne, S., Plastria, F.: 2-class Eigen Transformation Classification Trees. In: *Proceedings of KDIR 2009* (2009)
4. Elmqvist, N., Dragicevic, P., Fekete, J.-D.: Rolling the Dice: Multidimensional Visual Exploration using Scatterplot Matrix Navigation. *IEEE Transactions on Visualization and Computer Graphics* 14(6), 1148–1539 (2008)
5. Fisher, R.A.: The Use of Multiple Measurements in Taxonomic Problems. *Annals of Eugenics* 7, 179–188 (1936)
6. Heer, J., Robertson, G.: Animated Transitions in Statistical Data Graphics. *IEEE Transactions on Visualization and Computer Graphics* 13(6), 1240–1247 (2007)
7. Jolliffe, I.T.: *Principal Component Analysis.* Springer, Berlin (1986)
8. Plastria, F., De Bruyne, S., Carrizosa, E.: Dimensionality Reduction for Classification: Comparison of Techniques and Dimension Choice. In: Tang, C., Ling, C.X., Zhou, X., Cereone, N.J., Li, X. (eds.) *ADMA 2008. LNCS (LNAI), vol. 5139,* pp. 411–418. Springer, Heidelberg (2008)

An Optimised Algorithm to Tackle the Model Explosion Problem in CTL Model Update

Yulin Ding¹ and David Hemer²

¹ Airborne Mission Systems, Air Operations Division,
Defence Science and Technology Organisation,
PO Box 1500 Edinburgh, SA, 5111, Australia
yulin.ding@dsto.defence.gov.au

² School of Computer Science, The University of Adelaide
SA, 5005, Australia
david.hemer@adelaide.edu.au

Abstract. Computational Tree Logic (CTL) model update is an approach to software verification and modification, where minimal change is employed to generate updated models that represent the corrected software design. In this paper, we propose a new update principle named minimal change with maximal reachable states (II) which is a further optimisation of an existing algorithm to solve a model explosion problem during CTL model update. We provide comparison of the two methods based on Graph Theory. The algorithm of this update principle is also provided. Our experimental results show that in the case of updating the Andrew File System protocol model, the new CTL update approach significantly narrows down the committed models to fewer strong committed models.

Keywords: model checking, model update, minimal change.

1 Introduction

Error repairing is an formal-based approach that complements model checking. An example is the application of AI techniques to model checking and error repairing [1]. We have recently developed a software error repairing technique [6] for updating models expressed using CTL notation. The technique, referred to as *CTL model update*, is supported by a prototype algorithm and has been applied to several examples [5]. The methodology of model update unifies model checking and modification and can closely retain the efficiency of model checking as well as being able to develop a systematic approach for system modification. The CTL model update algorithms described in earlier papers typically generate multiple solutions, some less appropriate than others: we shall refer to this as the *model explosion problem*. The challenge is to minimise the number of non-optimal solutions to improve the efficiency of the model updater.

2 CTL Model Update: An Overview

2.1 CTL Syntax and Semantics

Definition 1. [3] Let AP be a set of atomic propositions. A Kripke model M over AP is a three tuple $M = (S, R, L)$ where: 1. S is a finite set of states, 2. $R \subseteq S \times S$ is a transition relation, 3. $L : S \rightarrow 2^{AP}$ is a function that assigns each state with a set of atomic propositions.

Definition 2. [7] Computation tree logic (CTL) has the syntax given in Baekus naur form: $\phi ::= \top \mid \perp \mid [p \mid (\neg\phi) \mid (\phi_1 \wedge \phi_2) \mid (\phi_1 \vee \phi_2) \mid \phi_1 \rightarrow \phi_2 \mid AX\phi \mid EX\phi \mid AG\phi \mid EG\phi \mid AF\phi \mid EF\phi \mid A[\phi_1 \cup \phi_2] \mid E[\phi_1 \cup \phi_2]]$ where p is any propositional atom.

Definition 3. [7] $M = (S, R, L)$ is a Kripke model for CTL and $s \in S$. A CTL formula ϕ holding in state s is denoted by $(M, s) \models \phi$. The satisfaction relation \models is defined by structural induction on CTL formulae:

1. $(M, s) \models p$ iff $p \in L(s)$.
2. $(M, s) \models \neg\phi$ iff $(M, s) \not\models \phi$.
3. $(M, s) \models \phi_1 \wedge \phi_2$ iff $(M, s) \models \phi_1$ and $(M, s) \models \phi_2$.
4. $(M, s) \models \phi_1 \vee \phi_2$ iff $(M, s) \models \phi_1$ or $(M, s) \models \phi_2$.
5. $(M, s) \models \phi_1 \rightarrow \phi_2$ iff $(M, s) \not\models \phi_1$, or $(M, s) \models \phi_2$.
6. $(M, s) \models AX\phi$ iff for all s_1 such that $(s, s_1) \in R$, $(M, s_1) \models \phi$.
7. $(M, s) \models AG\phi$ iff for all paths $\pi = [s_0, s_1, s_2, \dots]$, where $s_0 = s$ and $\forall s_i, s_i \in \pi$, $(M, s_i) \models \phi$.
8. $(M, s) \models A[\phi_1 \cup \phi_2]$ iff for all paths $\pi = [s_0, s_1, s_2, \dots]$, where $s_0 = s$, $\exists s_i \in \pi$, $(M, s_i) \models \phi_2$, and for each $j < i$, $(M, s_j) \models \phi_1$.

A CTL formula ϕ is evaluated on a Kripke model M and is satisfiable. A path in M from a state s is an infinite sequence of states

$$\pi \stackrel{def}{=} [s_0, s_1, \dots, s_{i-1}, s_i, s_{i+1}, \dots, s_j, \dots]$$

such that $s_0 = s$, $(s_i, s_{i+1}) \in R$ holds for all $i \geq 0$, $(s_i, s_{i+1}) \subseteq \pi$ and $s_i \in \pi$. We denote $s_i < s_j$ if s_i is a state earlier than s_j in π . We denote state s' as *succ*(s) if there is a relation (s, s') in R . s' could be one of a set of successor states of s . If $succ(s) \not\models \phi$, we express it as $succ(s, \neg\phi)$. If a state is accessible by transitions from an initial state s_0 , it is called a reachable state. We use $RS(\mathcal{M}) = RS(M, s_0)$ to denote the set of all reachable states from s_0 in M . Similarly, we use $RS^\beta(\mathcal{M}) = RS(M, s_i)$ to denote the set of reachable states from any state s_i in \mathcal{M} . The *unchanged reachable states* mean that the reachable states in an updated model are also in the original model. A state s is called true state for ϕ if $s \models \phi$ and called false state for ϕ if $s \not\models \phi$.

2.2 Minimal Change for CTL Model Update

Definition 4. [6] (CTL Model Update) Given a CTL Kripke model $M = (S, R, L)$ and a CTL formula ϕ such that $\mathcal{M} = (M, s_0) \not\models \phi$, where $s_0 \in S$. $Update(\mathcal{M}, \phi)$ derived from \mathcal{M} to satisfy ϕ results in an updated model $M' = (S', R', L')$ such that $\mathcal{M}' = (M', s'_0) \models \phi$ where $s'_0 \in S'$.

Update is achieved by applying a combination of primitive update operations PU1 to PU5. Given $M = (S, R, L)$, its updated model $M' = (S', R', L')$ is:

PU1: Adding a relation,

$S' = S, L' = L$, and $R' = R \cup \{(s_{ar}, s_{ar2})\}$ where $(s_{ar}, s_{ar2}) \notin R$ for $s_{ar}, s_{ar2} \in S$.

PU2: Removing a relation,

$S' = S; L' = L$, and $R' = R - \{(s_{rr}, s_{rr2})\}$ where $(s_{rr}, s_{rr2}) \in R$ for $s_{rr}, s_{rr2} \in S$.

PU3: Substituting a state and its associated relation(s),

$S' = S[s/s_{ss}]$ $R' = R \cup \{(s_i, s_{ss}), (s_{ss}, s_j) \mid (s_i, s), (s, s_j) \in R\} - \{(s_i, s), (s, s_j) \mid (s_i, s), (s, s_j) \in R\}$, and for all $s \in S \cap S'$, $L'(s) = L(s)$, and $L'(s_{ss})$ is a set of true variables assigned in s_{ss} .

PU4: Adding a state and its associated relation(s),

$S' = S \cup \{s_{as}\}$, $R' = R \cup \{(s_i, s_{as}), (s_{as}, s_j) \mid \text{for some } s_i, s_j \in S'\}$, and for all $s \in S \cap S'$, $L'(s) = L(s)$, and $L'(s_{as})$ is a set of true variables assigned in s_{as} .

PU5: Removing a state and its associated relation(s),

$S' = S - \{s_{rs} \mid s_{rs} \in S\}$, $R' = R - \{(s_i, s_{rs}), (s_{rs}, s_j) \mid \text{for some } s_i, s_j \in S\}$, and $L'(s) = L(s)$ for all $s \in S \cap S'$.

Given models $M = (S, R, L)$ and $M' = (S', R', L')$, where M' is an updated model from M by only applying operation PU_i on M . We define $Diff_{PU_i}(M, M') = Diff(R, R')$ ($i = 1, 2$), $Diff_{PU_i}(M, M') = Diff(S, S')$ ($i = 3, 4, 5$) and $Diff(M, M') = (Diff_{PU_1}(M, M'), \dots, Diff_{PU_5}(M, M'))$. For PU3 to update $s \not\models \phi$ to $s^* \models \phi$, we say $Diff(s, s^*)$ is minimal if we cannot find $Diff(s, s^*) \subseteq Diff(s, s^*)$, where $s'' \models \phi$.

Definition 5. (Closeness Ordering) Given three CTL Kripke models M, M_1 and M_2 , where M_1 and M_2 are obtained from M by applying PU1-PU5 operations, we say that M_1 is closer or as close to M as M_2 , denoted as $M_1 \leq_M M_2$, iff $Diff(M, M_1) \preceq Diff(M, M_2)$. We denote $M_1 <_M M_2$ if $M_1 \leq_M M_2$ and $M_2 \not\leq_M M_1$.

Definition 6. (Admissible Update) Given a CTL Kripke model $M=(S, R, L)$, $\mathcal{M} = (M, s_0)$, where $s_0 \in S$, and a CTL formula ϕ , $Update(\mathcal{M}, \phi)$ is called admissible if: (1) $Update(\mathcal{M}, \phi) = (M', s'_0) \models \phi$ where $M' = (S', R', L')$ and $s'_0 \in S'$; and (2) there does not exist another resulting model $M'' = (S'', R'', L'')$ and $s''_0 \in S''$ such that $(M'', s''_0) \models \phi$ and $M'' <_M M'$.

Theorem 1. $M = (S, R, L)$ is a Kripke model, $\mathcal{M} = (M, s_0)$ and $\mathcal{M} \not\models AG\phi$, where $s_0 \in S$ and ϕ is a propositional formula. An admissible model $\mathcal{M}' = Update(\mathcal{M}, AG\phi)$ can be obtained by the following: for each path $\pi = [s_0, \dots, s_i, \dots]$:

1. if for all $s < s_i$ in π , $s \models \phi$ but $s_i \not\models \phi$, PU2 is applied to s_i to remove relation (s_{i-1}, s_i) , or PU5 is applied to s_i to remove s_i and its associated relations, or
2. PU3 is applied to all states s in π not satisfying ϕ to substitute s with $s^* \models \phi$ and $Diff(s, s^*)$ is minimal.

Definition 7. (Minimal change with maximal reachable states(I)) [4,5]
 Given a CTL Kripke model $M = (S, R, L)$, $\mathcal{M} = (M, s_0)$ where $s_0 \in S$, and a CTL formula ϕ , $Update(\mathcal{M}, \phi)$ is called committed if the following conditions hold: (1) $Update(\mathcal{M}, \phi) = \mathcal{M}' = (M', s'_0)$ is admissible; and (2) there does not exist another resulting model $\mathcal{M}'' = (M'', s''_0)$ such that \mathcal{M}'' is admissible and $RS(\mathcal{M}) \cap RS(\mathcal{M}') \subset RS(\mathcal{M}) \cap RS(\mathcal{M}'')$.

3 A Further Optimisation

3.1 A New Approach: Reachable States from One State to Another

We consider an improvement of the reachable state principle in Definition 7. If two states are preserved in an update and there was a path between them in the original model, then there is still a path between them in the updated model. This improved reachable state principle in fact provides the reachability condition from all unchanged states in a model rather than only from initial states as described in Definition 7.

Definition 8. (Minimal change with maximal reachable states (II))
 Given a CTL Kripke model $M = (S, R, L)$, $\mathcal{M} = (M, s_0)$ where $s_0 \in S$, and a CTL formula ϕ , $Update(\mathcal{M}, \phi)$ is called strong committed if the following conditions hold: (1) $Update(\mathcal{M}, \phi) = \mathcal{M}' = (M', s'_0)$ is admissible or committed; and (2) there does not exist another resulting model $\mathcal{M}'' = (M'', s''_0)$ such that \mathcal{M}'' is admissible or committed and $RS^\beta(\mathcal{M}) \cap RS^\beta(\mathcal{M}') \subset RS^\beta(\mathcal{M}) \cap RS^\beta(\mathcal{M}'')$.

The strong committed update preserves all unchanged reachable states in an original model and preserves the reachability from any unchanged state to another after an update. The strong committed model results from the strong committed update. The total set of strong committed models are a subset of the total set of committed models. Thus, a constraint for deriving strong committed update instead of that for deriving committed update is added to Theorem 1.

3.2 Comparing Reachable State Principles Using Graph Theory

The reachable state principles in Definition 7 and 8 can be further analysed from a structural view using graph theory [2].

If the original model is a graph $G = (X, \Gamma)$, where X is the set of vertices and Γ is a mapping of the set X in X which shows how the vertices relate to each other. Its subgraph is $G_s = (X_s, \Gamma_s)$ with $X_s \subset X$; and for every $x_i \in X_s$, $\Gamma_s(x_i) = \Gamma(x_i) \cap X_s$. Thus, a subgraph has only a subset X_s of the set of vertices of the original graph but contains all the arcs whose initial and final vertices are both within this subset. The reachability matrix $R = [r_{ij}]$ is defined as follows:

$$r_{ij} = \begin{cases} 1 & \text{if vertex } x_j \text{ is reachable from vertex } x_i \\ 0 & \text{otherwise} \end{cases}$$

The reachable set of vertices from vertex x_i is:

$$R(x_i) = \{x_i\} \cup \Gamma(x_i) \cup \Gamma^2(x_i) \cup \dots \cup \Gamma^p(x_i),$$

where p is the cardinality of the reachable path from x_i .

Given two matrices $A = [a_{ij}]$ and $B = [b_{kl}]$, where $i \geq k, j = l$, if for any two elements a and b in identical positions of the two matrices $a \geq b$ holds, then we say $A \geq B$.

A Kripke model $M = (S, R, L)$ is mapped into a graph $M = (S, R)$, where S is the set of vertices and R is the set of edges. After model update, $M' = (S', R')$, where $S' = S_{unchange} \cup S_{update}$. $S_{unchange}$ is a set of unchanged states such that $S_{unchange} \subset S$. S_{update} is a set of updated states and $S_{update} \subset S'$. Before update, a subgraph of M containing all unchanged states $S_{unchange}$ and the set of states being updated with PU3 operation only S_{PU3} is $G = (S_{unchange} \cup S_{PU3}, R_u)$, where $S_{PU3} \subset S$ and $R_u \subset R$. After update, a subgraph of M' containing all unchanged states $S_{unchange}$ and the set of states derived from update by using PU3 operation only S'_{PU3} is $G' = (S_{unchange} \cup S'_{PU3}, R'_u)$, where $S'_{PU3} \subset S_{update}$ and $R'_u \subset R'$. From a graph theory view, the set of vertices $S_{PU3} = S'_{PU3}$ ¹.

For the subgraph G , reachability matrix is $RE = [r_{ij}]$, where i and j range over the number of states in G . We use $RE^{initial} = [r'_{(initial)j}]$, where $initial$ is the number of any initial states, to denote the reachability after update described in Definition 7. In Definition 7, reachability is only checked from initial states corresponding to roots in Graph theory. Also, $i \geq initial$. Thus, $RE \geq RE^{initial}$. After update optimised by using Definition 8, $RE^{AnyTwo} = [r'_{(AnyTwo)j}]$, where $AnyTwo$ is the number of any unchanged states and the number of updated states derived from using PU3 operation. It is obvious $RE = RE^{AnyTwo}$ under the definition of reachability matrices in [2], if there is not any unchanged reachable state lost during the update. Therefore, $RE^{AnyTwo} \geq RE^{initial}$. This proves that minimal change constrained with Definition 8 retains more unchanged reachable states than that of Definition 7 does during an update.

3.3 An Improved Algorithm

We have developed an alternative algorithm satisfying Theorem 1 and constrained with Definition 8 to derive strong committed models for optimising AG update. A Kripke model is $M = (S, R, L)$ and $\mathcal{M} = (M, s)$, where $s \in S$. \mathcal{M} is required to satisfy a propositional formula ϕ . The updated model of \mathcal{M} is $M' = (S', R', L')$ and $\mathcal{M}' = (M', s)$. RE and RE^{AnyTwo} are as described in Section 3.2.

Update_{AG}(\mathcal{M}, ϕ) /* $\mathcal{M} \not\models AG\phi$. Update \mathcal{M} to satisfy $AG\phi$. */
 { if $\mathcal{M}_0 = (M, s_0) \not\models \phi$, then PU3 is applied to s_0 ; else

- (1) applying PU3 on all $s_i \not\models \phi$ in \mathcal{M} ;
- (2) select a path $\pi = [s_0, s_1, \dots]$, where $\exists s \in \pi$, such that $\mathcal{M}_i = (M, s) \not\models \phi$;
 select the earliest state $s_i \in \pi$ such that $(M, s_i) \not\models \phi$;
 perform one of the following three operations:

¹ The states before and after update using PU3 are supposed to be the same because we do not consider the variables in states from the graph theory view.

```

(2.1) applying PU2 to remove relation  $(s_{i-1}, s_i)$  or
(2.2) applying PU5 to remove state  $s_i$  and its associated relations,
      obtain result  $\mathcal{M}'$  only if  $RE = RE^{AnyTwo}$ , else
(2.3) applying PU3 on all  $s_i \not\models \phi$  in  $\pi$ ;
if  $\mathcal{M}' \models AG\phi$ , return  $\mathcal{M}'$ ; else return  $\{Update_{AG}(\mathcal{M}', \phi)\}$ ;
}

```

After an update, the updated model is repeatedly checked whether it satisfies the required property $AG\phi$. If it does not, the function $Update_{AG}$ recursively calls itself until the updated model satisfies the specification property. The final updated model is a strong committed model.

The algorithm has been applied to Andrew File System 1 [5,8]. The number of strong committed models for this case is 125 which is narrowed down from 225 committed models. The whole process of the reachable state algorithm has been simulated in C code in our model updater prototype. Our model updater prototype automatically perform the algorithm and the output are strong committed models.

For other CTL formula update such as AX and AU which have the possibility of losing reachable states, their update algorithms are also constrained by Definition 8 in a similar format as that of AG. We have implemented the reachable state algorithm in code to embed the algorithm into the model updater protocol.

Acknowledgements

This work was funded in part by ARC Discovery Project DP0664478. We would like to acknowledge Yan Zhang and Lachlan Groenveld for their useful input. We also thank the Long Range research Team at AMS, AOD, DSTO for their support.

References

1. Bucaafurri, F., Eiter, T., Gottlob, G., Leone, N.: Enhancing model checking in verification by AI techniques. *Artificial Intelligence* 112, 57–104 (1999)
2. Christofides, N.: *Graph Theory – An Algorithmic Approach*. Academic Press, London (1975)
3. Clarke Jr., E., Grumberg, O., Peled, D.A.: *Model Checking*. The MIT Press, Cambridge (1999)
4. Ding, Y., Zhang, Y.: A Study of the Model Explosion Problem in CTL Model Update. In: *Proc. of the 20th International Conference on Software Engineering and Knowledge Engineering, SEKE 2008* (2008)
5. Ding, Y.: *Model Update for System Modifications*. Ph.D Thesis. School of Computing and Mathematics, University of Western Sydney (2007)
6. Ding, Y., Zhang, Y.: CTL model update: Semantics, computations and implementation. In: *Proc. of the 17th European Conference on Artificial Intelligence, ECAI 2006* (2006)
7. Huth, M., Ryan, M.: *Logic in Computer Science: Modelling and Reasoning about Systems*. University Press, Cambridge (2000)
8. Wing, J., Vaziri-Farahani, M.: A case study in model checking software. In: *Proc. of 3rd ACM SIGSOFT Symposium on the Foundations of Software Engineering* (October 1995)

Exploiting Symmetry in Relational Similarity for Ranking Relational Search Results

Tomokazu Goto, Nguyen Tuan Duc, Danushka Bollegala, and Mitsuru Ishizuka

The University of Tokyo, Japan

{goto,duc}@mi.ci.i.u-tokyo.ac.jp,

danushka@iba.t.u-tokyo.ac.jp, ishizuka@i.u-tokyo.ac.jp

Abstract. Relational search is a novel paradigm of search which focuses on the similarity between semantic relations. Given three words (A, B, C) as the query, a relational search engine retrieves a ranked list of words \mathcal{D} , where a word $D \in \mathcal{D}$ is assigned a high rank if the relation between A and B is highly similar to that between C and D . However, if C and D has numerous co-occurrences, then D is retrieved by existing relational search engines irrespective of the relation between A and B . To overcome this problem, we exploit the symmetry in relational similarity to rank the result set \mathcal{D} . To evaluate the proposed ranking method, we use a henchmark dataset of Scholastic Aptitude Test (SAT) word analogy questions. Our experiments show that the proposed ranking method improves the accuracy in answering SAT word analogy questions, thereby demonstrating its usefulness in practical applications.

Keywords: relational search, relational similarity, symmetry.

1 Introduction

Relational search is a novel search paradigm based on relational similarity of word pairs. For the query $\{(A,B),(C,?)\}$, in which $A, B,$ and C are input words, a relational search engine finds the words D such that the relation between A and B is also held between C and D . A candidate answer D is assigned a high rank when the word pair (C, D) has a high degree of relational similarity with the word pair (A, B) . In previous methods for relational search [3] and relational similarity measure [1], the relation between two words in a word pair is represented by lexico-syntactic patterns that frequently co-occur with those words. However, this approach imposes a bias towards the frequency of a word – a high frequency word D has a higher probability of being assigned a top rank, irrespective of the semantic relation shared between (A, B) and (C, D) . We propose a ranking method which uses the symmetry in relational similarity to alleviate this phenomenon.

To demonstrate the proposed ranking method, let us consider the query $\{(Google, Eric Schmidt), (Microsoft, ?)\}$. Here, “?” denotes an entity. *Steve Ballmer* is expected to be ranked at the top of the result list for this query because *Steve Ballmer* is the CEO of *Microsoft*, whereas *Eric Schmidt* is the CEO of *Google*. Moreover, when we use the inverse query $\{(Eric Schmidt, Google), (?, Microsoft)\}$, *Steve Ballmer* is also expected to be ranked as the first result. This is because relational similarity is invariant if both

word pairs are inverted [4]. The invariance of relational similarity under a symmetric transformation of word pairs provides us with a practical method to rank candidates in a relational search engine: we can obtain a better ranking if we take into account the ranking in the inverse query's result list.

In addition, we propose “*complementary rank*” for improving the precision in ranking the result set of a relational search query. When D is assigned a high rank (i.e., top rank) in the query $\{(A, B), (C, ?)\}$, we can expect that C is also assigned a high rank in the query $\{(A, B), (?, D)\}$. Therefore, we can consider the rank of C in the query $\{(A, B), (?, D)\}$ as an additional criterion for ranking D in the query $\{(A, B), (C, ?)\}$. We call this additional criterion as the “*complementary rank of D* ”. In the proposal method, we combine the symmetric property and complementary rank to improve the initial ranking.

2 Related Work

The idea of relational search has been introduced in Veale [6] and Bollegala, et al. [1]. Kato, et al. first implemented **relational search** [3] by issuing queries to a keyword-based Web search engine. To extract candidate answers, they first query a Web search engine for terms or lexico-syntactic patterns that are likely to appear only in documents which contain both A and B . The extracted term or pattern set T is supposed to contain terms or lexical patterns that express the relations between A and B . Then, they use C and a term $t \in T$ to find documents that contain both C and t . The candidate answer set \mathcal{D} is then defined as the set of terms that are likely to appear only in those documents. Then, they rank the candidate set using the likelihood of co-occurrence of the term D with the pair (C, t) . Our method also uses lexico-syntactic pattern to express the relations between A and B . However, the pattern generation algorithm and the scoring scheme are different. In particular, they use only the words in the mid-fix between A and B for extracting lexical patterns that might represent relations between A and B . On the other hand, we use wildcards and an n-gram model which can precisely capture the relation between A and B [1].

Bunescu and Mooney proposed an approach for overcoming the problem of bias due to high frequency words as mentioned in previous section [2]. However, their method needs a large amount of texts from Web documents for compute word frequencies. This can not be accomplished by using only snippets from a keyword-based Web search engine's results.

3 Method

To answer the query $\{(A, B), (C, ?)\}$, the proposed method first extracts lexical patterns that represent relations between A and B . The lexical patterns are n-grams of the context surrounding the pair (A, B) in a sentence. It then uses the keyword C along with these patterns to query a Web search engine for the answer D , similar to [3]. To improve the ranking of the results that are returned by the above procedure, we use the symmetry of relational similarity and complementary rank.



Fig. 1. Relational Search on the Web

Stem pair	ostrich	bird
1	lion	cat
2	goose	flock
3	ewe	sheep
4	cub	bear
5	primate	monkey

Fig. 2. An example SAT analogy question

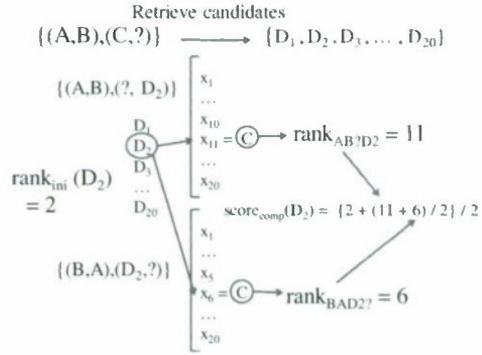


Fig. 3. Scoring candidates \mathcal{D} retrieved for the query $\{(A,B),(C,?)\}$

3.1 Relational Search on the Web

Fig. 1 shows the process to find the answer for the query $\{(A, B), (C, ?)\}$. First, we extract the semantic relation between A and B by issuing queries of type “ $A * * * B$ ” to a Web search engine¹ to obtain some text snippets that include A and B separated by up to three words. Here, “*” denotes a wildcard for any word. To increase the similarity between two pairs that have similar contexts, we generate all n -grams ($n \leq 5$) which contain both two words in a word pair as lexical patterns for the pair. For instance, in the sentence “*big A such as B is considered to be ...*”, we generate sequences such as “*big A such as B*”, “*A such as B*” and “*A such as B is*”. We obtain the lexical patterns by replacing A with the variable α and B with β in the original sub-sequences: “*big α such as β* ”, “ *α such as β* ” and “ *α such as β is*”. To avoid noisy patterns, we ignore all patterns whose frequencies are smaller than a frequency threshold ξ . We denote the set of these patterns by P .

To get candidate answers, for each pattern $p \in P$ we input the query “ $p[C/\alpha, */\beta]$ ” (including the double quotes) to the search engine. The formula $p[C/\alpha]$ represents the substitution of α by C in the pattern p . For this query, the search engine returns snippets which include C and other words in the pattern p and some extra words in this order. For example, for the query “lion is a large *”, the search engine returns snippets such as “lion is a large cat ...” or “lion is a large four-legged animal ...”. Because we want to get the word at the position of the wildcard * in the query, we add the those extra words into the candidate answer set \mathcal{D} . We then rank the a candidate $D \in \mathcal{D}$ using the following ranking score:

$$\text{score}_{\text{init}}(D) = \frac{\sum_{p \subseteq P_D} (\text{freq}("p[C/\alpha, D/\beta]"))}{\text{freq}("C * * * D")}. \tag{1}$$

In Formula 1, P_D are the patterns that appeared with D , $\text{freq}("p[C/\alpha, D/\beta]")$ is the frequency of co-occurrences of the word D with the word C and other words in the

¹ Yahoo Boss API <http://developer.yahoo.com/search/boss/>

patterns. Because the number of words between C and D is less than three, we normalize the sum by dividing the sum of $\text{freq}(\text{"p}[C/\alpha, D/\beta]\text{"})$ by the hit count of the query " C *** D ". Finally, we assign a rank to each $D \in \mathcal{D}$ using the score in Formula 1. We call this ranking as the **initial ranking**. The ranking score $\text{score}_{\text{init}}(D)$ is called the initial ranking score.

3.2 Symmetry in Relational Similarity

In the initial ranking, a candidate D might receive a top rank merely because it frequently occurs with C irrespective of the relation between A and B . To solve this problem, we propose a ranking score using the symmetry in relational similarity. Let us denote the relational similarity between (A, B) and (C, D) by $R((A, B), (C, D))$. Relational similarity will remain unchanged under certain permutations of the four words (e.g., $R((A, B), (C, D)) = R((B, A), (D, C))$). Therefore, the candidates that are ranked at the top by one form of the query (e.g., $(A, B), (C, ?)$) must also be ranked at the top by the other (alternative) forms of the query (e.g., $(B, A), (? , C)$). In other words, if D is an incorrect candidate, then it will be ranked at the top only in a small number of alternative forms of the query and it will receive bad ranks in almost all alternative forms. To consider the symmetric property, we define the score of D as follows:

$$\text{score}(D) = \frac{\text{score}_{\text{comp}}(D) + \text{score}_{\text{compR}}(D)}{2}. \quad (2)$$

In the above formula, $\text{score}_{\text{comp}}(D)$ is the score of D in the query $\{(A, B), (C, ?)\}$ when we take into account the complementary rank (we will explain complementary rank in the next section). Similarly, $\text{score}_{\text{compR}}(D)$ is the score of D in the other forms of the query whose similarities are invariant to a symmetric transformation (e.g., $\{(B, A), (? , C)\}$).

In addition to symmetry, we use complementary rank of C or D to rank candidate answers in a relational search engine. The complementary rank of a candidate D in the query $\{(A, B), (C, ?)\}$ is the initial rank of C in the query $\{(A, B), (? , D)\}$ and vice versa. We define the score of D by using complementary rank as follows,

$$\text{score}_{\text{comp}}(D) = \frac{\text{rank}_{\text{ini}}(D) + \frac{\text{rank}_{AB?D}(C) + \text{rank}_{BAD?}(C)}{2}}{2}, \quad (3)$$

where $\text{rank}_{\text{ini}}(D)$ is the rank of D in the initial ranking (i.e., ranking by $\text{score}_{\text{init}}(D)$ as shown in Formula 1), $\text{rank}_{AB?D}(C)$ is the initial rank of C in $\{(A, B), (? , D)\}$ and $\text{rank}_{BAD?}(C)$ is the initial rank of C in $\{(B, A), (D, ?)\}$. We denote the score of D in initial ranking of $\{(A, B), (C, ?)\}$ as $\text{score}_{\text{comp}}(D)$ and the score of D in initial ranking of $\{(B, A), (? , C)\}$ as $\text{score}_{\text{compR}}(D)$. By combining the Formula 2 and 3, we obtain the final score of D ($\text{score}(D)$) for ranking candidates $D \in \mathcal{D}$.

We illustrate the process of calculating $\text{score}_{\text{comp}}(D)$ in Figure 3 in the query $\{(A, B), (C, ?)\}$. We assign D a high rank if C is assigned high ranks when we use the queries $\{(A, B), (? , D)\}$ and $\{(B, A), (D, ?)\}$.

4 Evaluation

4.1 Experiments

To evaluate the proposed ranking algorithm, we use the SAT dataset [1,5]. The SAT dataset contains 374 word analogy questions selected from the Scholastic Aptitude Test.

Each questions has a question word pair (stem pair) and five choices for answer word pairs, in which the correct pair has the highest similarity with the stem pair as shown in Fig. 2. Therefore, we use the following method for solving SAT analogy questions.

Calculating the score of a word in the search result set

Given a stem word pair (A,B) and a choice word pair (C,D) (e.g., A is *ostrich*, B is *bird*, C is *lion* and D is *cat*), we first perform the query $\{(A,B),(C,?)\}$ to obtain a candidate answer set \mathcal{D} . Using the Formula 1, we rank the set \mathcal{D} to get the initial ranking. Suppose that the rank of D in this ranking is N^D . Next, we perform the query $\{(A,B),(?,D)\}$ to obtain a candidate set and record the rank (according to the score in Formula 1) N_1^C of C in this set. Similarly, we use the query $\{(B,A),(D,?)\}$ to get the rank of C as N_2^C . Finally, we define the SAT candidate score of D using the following formula:

$$SATSubScore(D) = \frac{N^D + \frac{N_1^C + N_2^C}{2}}{2} \quad (4)$$

Score of a SAT candidate answer

We calculate the score of a SAT candidate word pair $c = (C, D)$ as follow

$$SATScore(c) = \frac{SATSubScore(C) + SATSubScore(D)}{2} \quad (5)$$

After calculating SATScore for each candidate SAT answer, we select the choice whose score is minimal as the answer to the SAT question. To evaluate the performance, we compare the answer that our system outputs with the correct answer.

4.2 Results

We obtain 105 correct answers before using the symmetry and complementary rank. After using symmetry and complementary rank, we get 114 correct answers. Table 1 shows the experimental results. When we do not retrieve the word C or D for all five choices, we can not use the queries $\{(A,B),(C,?)\}$ or $\{(A,B),(?,D)\}$ respectively. In such cases, we can not estimate our method's effect, so we also measure the performance when we ignore those cases. After eliminating such cases, only 243 questions remain. For those questions, the proposed method achieved an accuracy of 46.9% when use the symmetry, whereas in initial ranking it is only 43.0%.

To measure our method's effect, we consider questions including correct answers and two or more answer candidates which include C or D . This results in 216 questions in which we made 78 correct answers (36.1%) before utilizing symmetry and complementary rank and 87 correct answers (40.3%) after. Therefore, by using symmetry and complementary rank, we could obtain 4.2% improvement in the SAT result.

Table 1. Comparison of correct rates

Criterion	Initial ranking	Using symmetry and complementary rank
# correct / # questions (recall)	28.1%	30.5%
# correct / # questions that we can get C or D (precision)	43.0%	46.9%
# correct / # questions that we can retrieve the correct choice and at least one other choice	36.1%	40.3%

5 Discussion

We observe that the use of symmetry and complementary rank improves the initial ranking. This shows that the proposed ranking method can be effectively applied to rank relational search results. Especially, the proposed method of exploiting symmetry of relations can be combined with advanced lexical pattern extraction techniques (e.g., PrefixSpan algorithm, etc.) to drastically improve the precision of relational search. Furthermore, one can improve the precision by combining existing relational search scoring algorithm such as [3] with the proposed scoring algorithm. Therefore, the proposed method can be smoothly integrated with other existing methods for ranking relational search results. The integration can be done easily because the proposed method exploits a special aspect of relations (i.e., the symmetry of relations) that is not utilized in existing approaches. It is worth noting that relational search is the first task concerning relational similarity in which complementary rank can be exploited and therefore be invented. In other tasks such as similarity measuring [1,5], complementary rank does not appear because in those tasks, the four words in the two pairs (A, B) and (C, D) are all given. On the other hand, in relational search or tasks in which one or more words are not given, we can define complementary rank to represent the strength of the relation between the candidate word and the input query word.

It is worth noting that the evaluation using SAT benchmark gives an interesting criterion for evaluating performance of a relational search engine, which can not be easily evaluated using normal criteria such as F-score or MRR (mean reciprocal rank).

6 Conclusion

We implemented relational search by using web search engine and proposed a ranking method for relational search. There are some noisy candidate words in the initial ranking of relational search results. To eliminate noisy candidate words from the initial ranking, we used a symmetric property and complementary rank. By using these features, we could improve 4.2% of precision. This shows that our proposed method of using symmetric property is effective for improving correct rate on SAT dataset and ranking relational search results.

References

1. Bollegala, D., Matsuo, Y., Ishizuka, M.: Measuring the similarity between implicit semantic relations from the web. In: Proc. of WWW 2009, pp. 651–660 (2009)
2. Bunescu, R.C., Mooney, R.: Learning to extract relations from the web using minimal supervision. In: Proc. of ACL 2007, pp. 576–583 (2007)
3. Kato, M.P., Ohshima, H., Oyama, S., Tanaka, K.: Query by analogical example: relational search using web search engine indices. In: Proc. of CIKM 2009, pp. 27–36 (2009)
4. Medin, D., Goldstone, R., Gentner, D.: Respects for similarity. *Psychological Review* 6(1), 1–28 (1991)
5. Turney, P., Littman, M., Bigham, J., Shnayder, V.: Combining independent modules to solve multiple-choice synonym and analogy problems. In: Proc. of RANLP 2003, pp. 482–486 (2003)
6. Veale, T.: The analogical thesaurus. In: Proc. of 15th Innovative Applications of Artificial Intelligence Conference (IAAI 2003), pp. 137–142 (2003)

Brain-Inspired Evolving Neuro-Fuzzy System for Financial Forecasting and Trading of the S&P500 Index

Weng Luen Ho, Whye Loon Tung, and Chai Quek

Centre for Computational Intelligence, Bloek N4 #2A-32,
School of Computer Engineering, Nanyang Technological University, Singapore 639798
stanley.ho@gmail.ntu.edu.sg, wltung@ntu.edu.sg,
ashcquek@ntu.edu.sg

Abstract. An interday financial trading system with a predictive model empowered by a novel brain-inspired evolving Mamdani-Takagi-Sugeno Neural-Fuzzy Inference System (eMTSFIS) is proposed in this paper. The eMTSFIS predictive model possesses synaptic mechanisms and information processing capabilities of the human hippocampus, resulting in a more robust and adaptive forecasting model as compared to existing econometric and neural-fuzzy techniques. The trading strategy of the proposed system is based on the moving-averages-convergence/divergence (MACD) principle to generate buy-sell trading signals. By introducing forecasting capabilities to the computation of the MACD trend signals, the lagging nature of the MACD trading rule is addressed. Experimental results based on the S&P500 Index confirmed that eMTSFIS is able to provide highly accurate predictions and the resultant system is able to identify timely trading opportunities while avoiding unnecessary trading transactions. These attributes enable the eMTSFIS-based trading system to yield higher multiplicative returns for an investor.

Keywords: evolving neural-fuzzy inference system, Mamdani-Takagi-Sugeno (MTS) fuzzy modeling, human hippocampus, time-series prediction, financial trading system, moving-averages-convergence/divergence (MACD), S&P500.

1 Introduction

The fundamental approach to financial trading is to identify movement trends and turning points, and subsequently make a decision to enter or exit the financial market. Generally, the investor will maintain an investment position until evidence indicates that the trend has reversed, of which, another decision will be made to take advantage of the trading opportunity that arises. Many investors rely on financial market analysis techniques, which can be broadly categorized as *fundamental analysis* and *technical analysis*, to formulate their trading decisions. Fundamental analysis focuses on the study of economic forces that affect supply and demand, for the purpose of forecasting the future price trends and deciding the long-term investment strategy [1]. In contrast, technical analysis bases its decision-making on historical financial data, such as price and volume [2]. Many financial theoreticians doubt the possibility of

using technical analysis to predict the financial market on the basis of the “Efficient Market Hypothesis” (EMH) [3], which imply that it is impossible to consistently outperform the market by using any information that is already available to the market. Despite the deeply entrenched beliefs of EMH, there has been substantial evidence [4, 5] on the predictability of financial markets using technical analysis.

This paper proposes the use of a brain-inspired incremental neuro-fuzzy system named the *evolving Mamdani-Takagi-Sugeno neuro-fuzzy inference system* (eMTSFIS) [6] to predict the financial market and to investigate the profitability of the derived trading system using historical data of the S&P500 market index.

2 eMTSFIS: The Evolving Mamdani-Takagi-Sugeno Neural-Fuzzy Inference System

The rule-generating procedure of the eMTSFIS model computationally mimics the human hippocampus, which is capable of a neurogenesis process [7] that has been regarded as the primary mechanism used to resolve the learning stability-plasticity dilemma in the human brain [8], via a recall comparator and novelty detection mechanism. Details of the rule-generating procedure of eMTSFIS are reported in [6]. In addition, the human hippocampus maintains its acquired knowledge using two primary synaptic mechanisms: *long-term potentiation* (LTP) [9] and *long-term depression* (LTD) [10]. LTP is responsible for the learning and reinforcement of memory traces in the hippocampal formation. LTD, on the other hand, is the mechanism for forgetting learnt information. Computationally, the eMTSFIS model mimics these neural mechanisms via the use of fuzzy rule potentials with the LTP and LTD concepts to construct a set of evolving IF-THEN Mamdani fuzzy rules to model non-stationary data generating processes [6].

The use of eMTSFIS to model financial trends allows a human investor to examine the inherent trend information extracted from the historical observations via highly interpretable fuzzy rules. Moreover, eMTSFIS can mitigate the effects of noise artifacts on the computed price predictions as it employs gaussian-shaped fuzzy sets to model (generalize) the characteristics of the past price movements. As such, a human trader will be able to develop a better understanding of the underlying characteristics of the observed price movements and make better and informed trading decisions to maximize his investment profits.

3 Financial Trading System Using Real-World Data

In this paper, a financial trading system with no predictive model and a financial trading system with eMTSFIS as the predictive model are introduced. In both systems, the trading signal at time t is represented by $F(t)$, where $F(t) \in \{1, -1\}$ with 1 and -1 representing the buy and sell signal respectively. The performances of the various trading strategies studied in this paper are defined by the portfolio terminal value $R(T)$ measuring the *wealth* created by the respective trading strategies using the notion of multiplicative returns [11] as shown in equation (1).

$$R(t) = \{1 + F(t-1)r(t)\} \{1 - \delta|F(t) - F(t-1)|\}, \quad t = 1, \dots, T \tag{1}$$

where $r(t) = (y(t)/y(t-1)) - 1$; the prices of the security being traded at time t and $(t-1)$ are denoted as $y(t)$ and $y(t-1)$ respectively; $F(t)$ is the action from a trading system at time t and is defined using equation (2) or (3); and δ is the transaction cost and is assumed to be a fraction of the transacted price value.

The financial trading system without a predictive model (TS-WOP) is shown in Figure 2. In this system, the trading signal $F(t)$ is derived using the MACD trading rule [2] of equation (2):

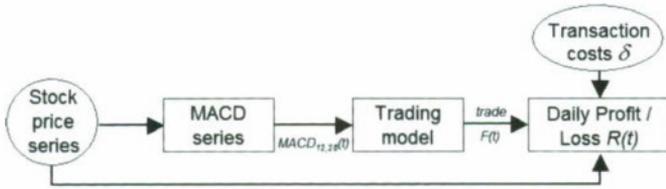


Fig. 2. The financial trading system using MACD with no predictive model

$$F(t) = \begin{cases} 1 & , \text{MACD}_{12,26} \text{ signal crosses above EMA}_9 \text{ trigger line} \\ -1 & , \text{MACD}_{12,26} \text{ signal falls below EMA}_9 \text{ trigger line} \\ F(t-1) & , \text{otherwise} \end{cases} \tag{2}$$

where $\text{MACD}_{12,26}$ denotes MACD employing the 12-days and 26-days exponential moving averages (EMA) [2] as the *fast* and *slow* signals respectively; and EMA_9 denotes the 9-days EMA of MACD that is used in place of the zero reference line as the trigger to generate the buy-sell trading signals.

The proposed financial trading system with eMTSFIS as a predictive model is shown in Figure 3. This system seeks to address the lagging nature of the MACD trading rule and to enhance its timeliness in spotting trading opportunities by introducing forecasting capabilities to the computation of the underlying trend signals. The three most recent daily closing prices [i.e., $\text{Closing}(t-2)$, $\text{Closing}(t-1)$, $\text{Closing}(t)$] are used as inputs for the eMTSFIS predictive model to forecast the future closing price at V days later [i.e., $\text{Closing}'(t+V)$]. The eMTSFIS predictive model is trained using supervised learning [6] on a set of historical S&P500 daily closing price training samples. The trained eMTSFIS is then employed to predict a set of out-of-sample closing levels. All the predicted closing prices are then fed into the trading model, which computes the predicted MACD' and generates the trading signal $F(t)$ using equation (3).

$$F(t) = \begin{cases} 1 & , \text{MACD}'_{12,26}(t+V) \text{ signal crosses above EMA}_9 \text{ trigger line} \\ -1 & , \text{MACD}'_{12,26}(t+V) \text{ signal falls below EMA}_9 \text{ trigger line} \\ F(t-1) & , \text{otherwise} \end{cases} \tag{3}$$

To demonstrate the forecasting capabilities of the proposed eMTSFIS model, the prediction results of eMTSFIS are benchmarked against two well-established evolving neural-fuzzy systems, i.e. EFuNN [12] and DENFIS [13], as well as two

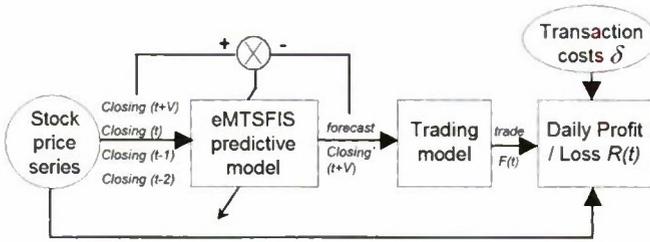


Fig. 3. The financial trading system with the eMTSFIS predictive model

econometric forecasting models, i.e. the autoregressive moving average (ARMA) [14] model and the Random Walk model [15]. Finally, the performance of the proposed eMTSFIS-based financial trading system shown in Figure 3 is benchmarked against those of a simple buy-and-hold strategy, a trading system with no prediction, a trading system with perfect prediction and two trading systems with EFuNN and DENFIS as the respective predictive model using historical data of the S&P500 market index.

All the aforementioned predictive models are constructed as 3-input-1-output systems configured with default parameters. For the trading systems employing a predictive model, the trading signals are generated using equation (3). Correspondingly, the trade signals for the trading system employing perfect predictions are also generated using equation (3) but the predicted $Closing'(t+V)$ prices are now replaced with the actual $Closing(t+V)$ prices. The final portfolio value of each benchmarked trading system is computed using equation (1), with the initial portfolio value $R(0) = 1.0$ and the transaction cost rate $\delta = 0.2\%$.

3.1 Forecasting and Trading of the S&P500 Market Index

In this experiment, the benchmarked trading systems are evaluated using the S&P500 market index. The experimental data is obtained from Yahoo Finance and consists of 15637 daily closing values spanning the period of 05 January 1950 to 11 December 2009. The training data set for the various predictive models consists of the initial 7500 index values while the out-of-sample data set contains the remaining 8137 index values. The three most recent daily closing index values are given as inputs to the various predictive models to forecast the closing index value five days later.

As observed from Table 1, the eMTSFIS predictive model has superior forecasting performance as compared to the econometric models (i.e. ARMA and Random Walk) and the evolving neural-fuzzy systems DENFIS and EFuNN. According to [16],

Table 1. Forecasting results of different predictive models on the S&P500 index

Predictive Model	Test Error (RMSE)	Number Of Rules
ARMA (1,1)	10.369	N.A.
Random Walk	9.9203	N.A.
DENFIS	0.7203	6
EFuNN	0.7313	213
eMTSFIS	0.3901	38

ARMA and Random Walk have poor forecasting results because of their linear structures and other inherent limitations. On the other hand, despite having a larger rule-base as compared to DENFIS, the Mamdani-type fuzzy rules identified by eMTSFIS are highly interpretable. This contrasts favourably to the TSK-type fuzzy rules in DENFIS, which are difficult to comprehend.

Table 2 shows the overall performances of the benchmarked trading systems as reflected by their portfolio end value $R(T)$ and the square of the Pearson correlation (SPC) between the actual and predicted index series. In Table 2, B&H denotes the buy-and-hold strategy; TS-WOP and TS-PP refer to the trading systems with no prediction and with perfect predictions respectively; TS-DENFIS, TS-EFuNN and TS-eMTSFIS are the respective trading systems employing DENFIS, EFuNN and eMTSFIS as the predictive model.

Table 2. Performances of the different trading systems using the S&P500 index

Trading System	$R(T)$	SPC
B&H	10.40	N.A.
TS – WOP	11.59	N.A.
TS – DENFIS	13.66	0.9763
TS – EFuNN	12.48	0.9044
TS – eMTSFIS	15.31	0.9958
TS – PP	59.57	1.0

As shown by the multiplicative returns generated for an investor employing the various trading strategies in Table 2, the trading system with the proposed eMTSFIS as a predictive model (TS-eMTSFIS) outperformed the simple buy-and-hold strategy, the trading system with no prediction and the trading systems employing DENFIS and EFuNN as predictive models. The superior performance of TS-eMTSFIS can be analyzed by inspecting its trading signals as shown in Figure 4. Based on region (a) of Figure 4, TS-eMTSFIS is able to enter into a long (buy) position at a lower price and at an earlier time than the trading system with no prediction (TS-WOP) due to an accurate forecast by the eMTSFIS model. Similarly in region (c) of Figure 4, TS-eMTSFIS is able to secure a short (sell) position at a higher price and at an earlier time than TS-WOP. These well-timed trades translate to a higher multiplicative return $R(T)$ as compared to other trading strategies shown in Table 2. In addition, the closing index values predicted by eMTSFIS have a higher correlation to the actual closing levels when benchmarked to DENFIS and EFuNN. This translates to improved decision making and enhances the timeliness of the trading system TS-eMTSFIS in spotting trading opportunities, thus contributing to a higher multiplicative return $R(T)$. Moreover, region (b) of Figure 4 showed that TS-eMTSFIS is able to avoid some unnecessary trading transactions, thus reducing the transaction costs incurred. This can be attributed to the ability of eMTSFIS to generalize the characteristics of the past index movements, thus mitigating the effects of noise artifacts on the computed MACD series that determines the corresponding trading signals.

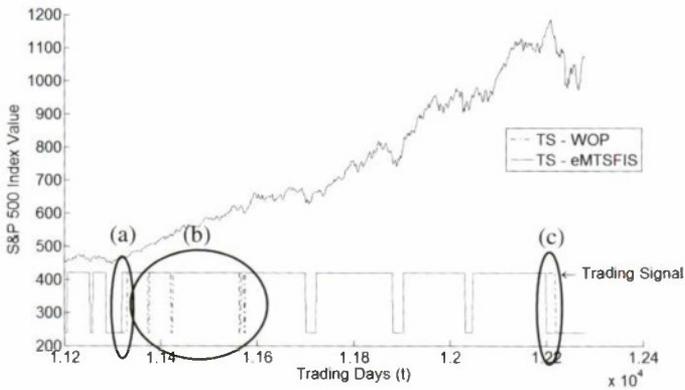


Fig. 4. Trading signals on S&P 500 index from time $t=11200$ to 12400

4 Conclusion

A financial trading system employing a novel brain-inspired evolving Mamdani-Takagi-Sugeno neuro-fuzzy inference system (eMTSFIS) predictive model is proposed. In this paper, eMTSFIS is used to model and forecast the daily closing index values of the S&P500 market index. Experimental results confirmed that eMTSFIS is able to provide highly accurate predictions and identify timely trading opportunities while avoiding unnecessary trading transactions. Collectively, these attributes enable the proposed eMTSFIS-based trading system to yield higher multiplicative returns for an investor.

References

1. Abarbanell, J.S., Bushee, B.J.: Fundamental analysis, future earnings, and stock prices. *J. Accounting Research* 35, 1–24 (1997)
2. Pring, M.J.: *Technical Analysis Explained: the Successful Investor's Guide to Spotting Investment Trends and Turning Points*, 4th edn. McGraw-Hill, New York (2002)
3. Fama, E.F.: Efficient capital markets: a review of theory and empirical work. *J. Finance* 25, 383–417 (1970)
4. Lo, A.W., Mamaysky, H., Wang, J.: Foundations of technical analysis: computational algorithms, statistical inference, and empirical implementation. *J. Finance* 55, 1705–1765 (2000)
5. Plummer, T., Ridley, A.: *Forecasting Financial Markets: The Psychological Dynamics of Successful Investing*, 4th edn. Kogan Page, London (2003)
6. Ho, W.L., Tung, W.L., Quek, C.: The Evolving Mamdani-Takagi-Sugeno based Neural-Fuzzy Inference System with Improved Interpretability-Accuracy. In: *FUZZ-IEEE 2010 (2010 IEEE World Congress on Computational Intelligence)*, Centre De Convencions International De Barcelona, Spain, July 18-23 (2010)
7. Kempermann, G., Wiskott, L., Gage, F.: Functional significance of adult neurogenesis. *Current Opinion of Neurobiology* 14, 186–191 (2004)

8. Wiskott, L., Raseh, M., Kempermann, G.: A functional hypothesis for adult hippocampal neurogenesis: Avoidance of catastrophic interference in the dentate gyrus. *Hippocampus* 16(3), 329–343 (2006)
9. Whitlock, J.R., Heynen, A.J., Shuler, M.G., Bear, M.F.: Learning induces long-term potentiation in the hippocampus. *Science* 313(5790), 1093–1097 (2006)
10. Bear, M.F., Abraham, W.C.: Long-term depression in hippocampus. *Annual Review of Neuroscience* 19, 437–462 (1996)
11. Moody, J., Wu, L., Liao, Y., Saffell, M.: Performance functions and reinforcement learning for trading systems and portfolios. *Journal of Forecasting* 17(5-6), 441–470 (1998)
12. Kasabov, N.: Evolving fuzzy neural networks for supervised/unsupervised online knowledge-based learning. *IEEE Trans. on Systems, Man and Cybernetics, Part B* 31(6), 902–918 (2001)
13. Kasabov, N., Song, Q.: DENFIS: Dynamic evolving neural-fuzzy inference system and its application for time-series prediction. *IEEE Transactions on Fuzzy Sys.* 10(2), 144–154 (2002)
14. Box, G.E.P., Jenkins, G.M., Reinsel, G.C.: *Time Series Analysis: Forecasting and Control*, 3rd edn. Prentice Hall, Englewood Cliffs (1994)
15. Malkiel, B.G.: *A Random Walk Down Wall Street*, 6th edn. W.W. Norton & Company, Inc., New York (1973)
16. Lin, C.S., Khan, H.A., Huang, C.-C.: Can the neuro fuzzy model predict stock indexes better than its rivals? CIRJE-F-165, Faculty of Economics. University of Tokyo (2002)

Bargain over Joint Plans^{*}

Wei Huang¹, Dongmo Zhang¹, Yan Zhang¹, and Laurent Perrussel²

¹ Intelligent Systems Laboratory, University of Western Sydney, Australia

² IRIT, Université de Toulouse, France

Abstract. This paper studies the problem of multi-agent planning in the environment where agents may need to cooperate in order to achieve their individual goals but they do so only if the cooperation is beneficial to each of them. We assume that each agent has a reward function and a cost function that determines the agent's utility over all possible plans. The agents negotiate to form a joint plan through a procedure of alternating offers of joint plans and side-payments. We propose an algorithm that generates an agreement for any given planning problem and show that this agreement maximizes the gross utility and minimizes the distance to the ideal utility point.

Keywords: multi-agent planning, joint plan, side-payment, bargaining.

1 Introduction

Multiagent planning has been an emerging research topic in recent years in the area of Artificial Intelligence [1,2,3,4,5,7]. Most existing studies on multiagent planning involve planning for common goals, plan coordinating, plan merging and synchronized planning. Most of the existing frameworks on multiagent planning are based either on the assumption that all agents have common goals therefore will be fully cooperative for a joint plan or on the assumption that all agents must reveal their private information, such as goals, rewards, costs and/or utilities, to other agents or arbitrators. In many real-world situations, none of the assumptions satisfies. It is a great challenge to find a joint plan for a multiagent system in which all agents are self-interested with individual goals and private information.

In this paper, we propose a solution to multiagent planning based on the following scenario:

- Each agent in the system has its own goals, reward of goal achievement and costs of actions.
- All agents are self-interested but profit-driven. An agent only concerns about its own goals. However, to attract other agents to join its plan, an agent may offer the other agents some payment (named side-payment) if the other agents agree on the joint plan.

^{*} This research was supported by the Australian Research Council through Linkage Project LP0777015.

- An agent can make a proposal of a plan with actions from the other agents or its own (therefore a joint plan) and a side-payment scheme. An agent can accept other agents' proposal if the net profit it receives from this plan (possible reward minus costs plus side-payment) surpasses any of its own plans, reject the proposal by making a counter proposal.

Based on the above scenario, we propose a planning procedure, named *Planning Procedure based on Bargaining* (PPB). The procedure is based on an alternating-offer model of bargaining for two-agent bargaining situations [8]. The planning procedure proceeds in several rounds. In each round, only one agent can make a proposal, which consists of a plan and a side payment scheme. If the other agent accepts the proposal, the procedure terminates and the current proposal becomes the final agreement; otherwise, it is the other agent's turn to make a proposal. We show that PPB is correct, complete, and terminating.

This paper is structured as follows. Firstly, we introduce some formal preliminaries to represent the planning problems. Secondly, we define the concept of plan proposals and bargaining mechanism. Thirdly, we propose a planning procedure based on the bargaining mechanism and show its properties. Finally, we discuss related work and future research directions.

2 Planning Domains and Problems

In this section we present a model of dynamic systems based on which the planning problems that will be dealt with in this paper is described.

A *multi-agent planning domain* is a tuple $\Sigma = \langle \mathcal{S}, s_0, \Phi, \mathcal{A}, \mathcal{T} \rangle$, where \mathcal{S} is a set of states, $s_0 \in \mathcal{S}$ is the initial state, Φ is a non-empty set of agents, \mathcal{A} is a set of actions, and $\mathcal{T} \subseteq \mathcal{S} \times \Phi \times \mathcal{A} \times \mathcal{S}$ represents the state transition relation. $\langle s, \varphi, a, s' \rangle \in \mathcal{T}$ means that φ can perform action a at state s and bring about s' as one of the possible result states.

For simplicity, we assume in this paper that $|\{s' \in \mathcal{S} : \langle s, \varphi, a, s' \rangle \in \mathcal{T}\}| \leq 1$ for each $\langle s, \varphi, a \rangle$ in $\mathcal{S} \times \Phi \times \mathcal{A}$, i.e., we only consider deterministic state transitions. All actions are assumed to be asynchronous, that is to say, at most one agent performs an action at each state.

Definition 1. Given a planning domain Σ , a plan π for Σ is a finite sequence in the form $\langle \varphi_1, a_1 \rangle; \langle \varphi_2, a_2 \rangle; \dots; \langle \varphi_n, a_n \rangle$, where $\varphi_i \in \Phi$ and $a_i \in \mathcal{A}$. The plan π is called to be applicable to Σ if there exist $s_1, s_2, \dots, s_n \in \mathcal{S}$ such that $\langle s_{i-1}, \varphi_i, a_i, s_i \rangle \in \mathcal{T}$ for all $0 < i \leq n$. s_n and n are referred to as the last state and the length of the plan, denoted by $\text{LSTATE}(\pi)$ and $\text{LENGTH}(\pi)$, respectively. $\text{AGTS}(\pi)$ denotes the set of agents that are involved in π , i.e., $\text{AGTS}(\pi) = \{\varphi \in \Phi : \varphi \text{ appears in } \pi\}$.

Given a planning domain, assume that each agent has its own goals, rewards if the goals are achieved and costs of actions. A multi-agent planning problem is to find a joint plan that can achieve the goals of all the agents meanwhile maximize their rewards and minimize their costs of actions.

Definition 2. A planning problem is a tuple $\mathcal{P} = \langle \Sigma, \mathcal{G}, r, c \rangle$, where

- $\Sigma = \langle \mathcal{S}, s_0, \Phi, \mathcal{A}, T \rangle$ is a planning domain.
- $\mathcal{G} : \Phi \rightarrow 2^{\mathcal{S}}$ is a goal function that specifies each agent's goal states.
- $r : \Phi \rightarrow \mathbb{Z}_+$ is a reward function that assigns to each agent a non-negative integer, representing the reward an agent can received if its goals are achieved.
- $c : \Phi \times \mathcal{A} \rightarrow \mathbb{Z}_+$ is a cost function that specifies the cost of each action to each agent.

Note that for every agent φ , $\mathcal{G}(\varphi)$, $r(\varphi)$, and $c_\varphi = c(\varphi, a)$ are φ 's private information. Therefore we write $\varphi.\mathcal{G}$, $\varphi.r$, and $\varphi.c$ instead of $\mathcal{G}(\varphi)$, $r(\varphi)$, and c_φ , respectively, to indicate that these functions are implemented in agent φ .

Given a planning problem \mathcal{P} , let $\Omega(\mathcal{P})$ denote the set of all the applicable plans for the planning domain of \mathcal{P} . For each agent $\varphi \in \Phi$ and $\pi = \langle \varphi_1, a_1 \rangle; \langle \varphi_2, a_2 \rangle; \dots \in \Omega(\mathcal{P})$, we define φ 's utility of π as follows:

$$u_\varphi(\pi) = \text{REW}_\varphi(\pi) - \sum_{i=1}^{\text{LENGTH}(\pi)} \text{COST}_\varphi(\varphi_i, a_i)$$

where $\text{REW}_\varphi(\pi) = \varphi.r$ if $\text{LSTATE}(\pi) \in \varphi.\mathcal{G}$; 0 otherwise and $\text{COST}_\varphi(\varphi_i, a_i) = \varphi.c(a_i)$ if $\varphi = \varphi_i$; 0 otherwise.

We use u_φ^\perp to denote the maximal value of utility that φ can achieve without other agent's involvement, i.e., $u_\varphi^\perp = \max_{\pi \in \Omega(\mathcal{P})} \{u_\varphi(\pi) | \text{AGTS}(\pi) \subseteq \{\varphi\}\}$. u_φ^\perp acts as φ 's bottom line for bargaining. In other words, φ is willing to cooperate with other agents only if the cooperation can bring to φ a utility value which is strictly greater than u_φ^\perp (individual rationality). Let $\Omega^\perp(\mathcal{P})$ be the set of plans which are individual rational, i.e., $\Omega^\perp(\mathcal{P}) = \{\pi \in \Omega(\mathcal{P}) | (\forall \varphi \in \Phi) u_\varphi(\pi) > u_\varphi^\perp\}$.

Similarly, we use u_φ^\top to denote the maximal utility the agent φ can gain with respect to the current planning situation provided all other agents are individual rational, i.e., $u_\varphi^\top = \max_{\pi \in \Omega^\perp(\mathcal{P})} u_\varphi(\pi)$. Indeed u_φ^\top is the ideal outcome of φ .

3 Bargaining Situation

To simplify the presentation of our approach, we will focus on two-agent planning problems, i.e., $\Phi = \{\varphi_{-1}, \varphi_1\}$. We call utility pair $(u_{\varphi_{-1}}^\top, u_{\varphi_1}^\top)$ the *ideal point*, denoted by $\text{IDP}(\mathcal{P})$. For any $j \in \{-1, 1\}$ and $\{\pi', \pi\} \subseteq \Omega^\perp(\mathcal{P})$, if $u_{\varphi_j}(\pi') > u_{\varphi_j}(\pi)$, then agent φ_j will prefer π' to π . If φ_{-j} does not agree to perform π' , then φ_j can propose a side payment such that the amount proposed to φ_{-j} is not greater than $u_{\varphi_j}(\pi') - u_{\varphi_j}(\pi) - 1$. If this proposal does not work, then φ_j must abandon π' and consider π instead.

Definition 3. A proposal to \mathcal{P} is a pair $p = \langle \pi, \xi \rangle$ such that π is a plan for the planning domain of \mathcal{P} , $\xi : \Phi \rightarrow \mathbb{Z}$ is a side payment function which satisfies $\sum_{\varphi \in \Phi} \xi(\varphi) = 0$. For any $k \in \mathbb{Z}$, ξ_k denotes the side payment function that assigns k to φ_1 , and $-k$ to φ_{-1} . For each $\varphi \in \Phi$, the utility of p to φ is defined as: $u_\varphi(p) = u_\varphi(\pi) + \xi(\varphi)$.

$\text{PRO}(\mathcal{P})$ denotes the set of possible proposals. Proposal $p = \langle \pi, \xi_k \rangle \in \text{PRO}(\mathcal{P})$ if and only if: (1) $\pi \in \Omega^\perp(\mathcal{P})$ and, (2) $u_{\varphi_{-1}}(p) > u_{\varphi_{-1}}^\perp$ and $u_{\varphi_1}(p) > u_{\varphi_1}^\perp$.

In order to reach an agreement (i.e., a proposal accepted by the two agents), the agents can bargain with each other by proposing proposals one by one. Once an agreement $p = \langle \pi, \xi \rangle$ is reached, all the agents in $\text{AGTS}(\pi)$ will cooperate to perform π , and the *gross utility*, i.e., $\sum_{\varphi \in \Phi} u_\varphi(\pi)$ will be redistributed among Φ such that each agent φ 's real income is $u_\varphi(p)$. For a proposal p to \mathcal{P} , we use $\text{DIS}(p) = \sqrt{(u_{\varphi_{-1}}^\top - u_{\varphi_{-1}}(p))^2 + (u_{\varphi_1}^\top - u_{\varphi_1}(p))^2}$ to denote the distance between $\text{IDP}(\mathcal{P})$ and the utility pair derived from p . In other words, $\text{DIS}(p)$ describes the concessions made by the two agents to achieve p . This leads to the notion of *solution* which characterizes the Pareto optimal proposals which entail minimal concessions.

Definition 4. *Proposal p is a solution to \mathcal{P} if it satisfies the following three conditions:*

Individual rationality: $p \in \text{PRO}(\mathcal{P})$;

Pareto optimality: *there is no proposal $p' \in \text{PRO}(\mathcal{P})$ such that $u_\varphi(p') > u_\varphi(p)$ for all $\varphi \in \Phi$;*

Minimal concession: $\text{DIS}(p) = \text{MIN}\{\text{DIS}(p') | p' \in \text{PRO}(\mathcal{P})\}$.

4 The Bargaining Mechanism

In this section, we present a planning procedure based on bargaining, and show its properties. The procedure is used for two-agent planning settings, in which all utility functions and goals are private information and cannot be revealed.

The planning procedure based on bargaining (PPB) is defined as follows.

step 1: Each agent $\varphi \in \Phi$ calculates the set of plans $bups_\varphi = \{\pi | (u_\varphi(\pi) > u_\varphi^\perp) \wedge (\text{LENGTH}(\pi) \leq \delta)\}^1$, and sends $bups_\varphi$ to an arbitrator φ^* .

step 2: φ^* calculates $\Omega^\perp(\mathcal{P}) = bups_{\varphi_{-1}} \cap bups_{\varphi_1}$. If $\Omega^\perp(\mathcal{P}) = \emptyset$, then φ^* announces the result of the procedure is *failure*, and the procedure stops. Otherwise, φ^* sets the set of plans to be considered $ps(0) := \Omega^\perp(\mathcal{P})$, $i := \text{RAND}(\{-1, 1\})^2$, sends $ps(0)$ and i to each $\varphi \in \Phi$.

step 3: Each $\varphi_j \in \Phi$ sets its proposal being considered $p_{\varphi_j}(0) := \langle \text{RAND}(pls_{\varphi_j}), \xi_0 \rangle$, where

$$pls_{\varphi_j} = \arg \max_{\pi \in ps(0)} u_{\varphi_j}(\pi),$$

and sends $p_{\varphi_j}(0)$ to φ_{-j} . Let $t := 0$, $\theta_{-1} := 0$, and $\theta_1 := 0$.

step 4: If $u_{\varphi_i}(p_{\varphi_{-i}}(t)) \geq u_{\varphi_i}(p_{\varphi_i}(t))$, then φ_i sends *done* to φ_{-i} , goto **step 7**. Otherwise φ_i sets $ps(t+1) := \{\pi \in ps(t) | u_{\varphi_i}(\pi) > u_{\varphi_i}(p_{\varphi_{-i}}(t))\}$, and φ_{-i} sets $p_{\varphi_{-i}}(t+1) := p_{\varphi_{-i}}(t)$.

¹ We adopt and, for ease of presentation further strengthen the *simple agents* assumption, requiring each plan to be bounded in length by a fixed δ .

² Given a set, RAND returns an element of the set randomly.

step 5: Suppose $p_{\varphi_i}(t) = \langle \pi, \xi \rangle$. If $ps(t + 1) = \emptyset$ or $u_{\varphi_i}(p_{\varphi_i}(t)) > \text{MAX}\{u_{\varphi_i}(\pi) \mid \pi \in ps(t + 1)\}$ then $\theta_i := \theta_i + 1$ and φ_i sets $p_{\varphi_i}(t + 1) := \langle \pi, \xi' \rangle$ such that $\xi'(\varphi_i) = \xi(\varphi_i) - 1$ and $\xi'(\varphi_{-i}) = \xi(\varphi_{-i}) + 1$. Otherwise φ_i sets $p_{\varphi_i}(t + 1) := (\text{RAND}(pls'_{\varphi_i}), \xi_0)$, where

$$pls'_{\varphi_i} = \arg \max_{\pi' \in ps(t+1)} u_{\varphi_i}(\pi').$$

step 6: φ_i sends $p_{\varphi_i}(t + 1)$ to φ_{-i} . Let $t := t + 1$ and $i := -i$. Return to step 4.

step 7: Suppose $p_{\varphi_{-i}}(t) = \langle \pi^*, \xi^* \rangle$. Then φ^* sets $j := \text{RAND}(\{-1, 1\})$, and announces $p = \langle \pi^*, \xi' \rangle$ is the result of the procedure, where $\xi'(\varphi_j) = \xi^*(\varphi_j) + \theta_{\varphi_j} - \lfloor 0.5 * w \rfloor^3$, $\xi'(\varphi_{-j}) = \xi^*(\varphi_{-j}) + \theta_{\varphi_{-j}} - \lceil 0.5 * w \rceil$, and $w = \theta_{\varphi_{-1}} + \theta_{\varphi_1}$.

If we observe this procedure, we remark that, for all $j \in \{-1, 1\}$, φ_j only sends proposals to φ_{-j} and φ^* . So for all $\pi \in \Omega^\perp(\mathcal{P})$, φ_{-j} and φ^* can not know $u_{\varphi_j}(\pi)$ (and of course, also $\varphi_j.g$, $\varphi_j.r$, and $\varphi_j.c$) during the procedure.

We now show the properties of PPB. The first key result states that PPB always terminates in polynomial time.

Theorem 1. *Under the simple agents assumption, PPB is guaranteed to terminate, and it is polynomial in $\min\{u_{\varphi_{-1}}^*, u_{\varphi_1}^*\}$, where $u_{\varphi_i}^* = u_{\varphi_i}^\top - \min\{u_{\varphi_i}(\pi) \mid \pi \in \Omega^\perp(\mathcal{P}) \text{ and } u_{\varphi_{-i}}(\pi) = u_{\varphi_{-i}}^\top\}$.*

The second property states that if there is a solution for the planning problem, then the proposed procedure will not fail.

Theorem 2. *failure is the result of PPB if and only if there is no solution to \mathcal{P} .*

The following theorem shows that the resulting proposal is a solution to \mathcal{P} .

Theorem 3. *If PPB returns $p \neq \text{failure}$, then p is a solution to \mathcal{P} .*

5 Conclusion and the Related Work

In this paper, we have proposed a model of multi-agent planning problems based on a bargaining mechanism. We have considered a class of planning situations in which each agent has its own goals, reward function and cost function. Agents bargain over joint plans with possible side payments. We have proposed a planning procedure which possesses the following properties: (1) the procedure always terminates in polynomial time; (2) for any given planning problem, if the set of individual rational plans is non-empty, the procedure can generate a joint plan at its termination; (3) the side payment associated with the resulting plan leads to a bargaining solution that is individual rational and Pareto optimal with minimal distance to the ideal point.

Most of the early work on multiagent planning is built up on fully cooperative multi-agent systems, such as the multi-entity model [7] and MA-STRIPS

³ $\lceil \cdot \rceil$ and $\lfloor \cdot \rfloor$ denote the ceil and floor function on real numbers, respectively.

planning [4]. Recently, game-theoretic approaches were applied to the problem of multiagent planning so that common plans or joint plans can be generated among self-interested agents [1,2]. In particular, Brafman *et al.* formalized a multiagent planning problem into a planning game which captures a rich class of planning scenarios [3]. However, these existing works on multiagent planning are based on either the assumption that all agents have common goals or the assumption that all agents must reveal their private information, such as goals, rewards, costs and/or utilities, to other agents or arbitrators. In contrast, our approach to multiagent planning is based on a bargaining mechanism, which assumes that goals, rewards and costs are private information and will not be revealed to any other agents or arbitrators. In fact, these pieces of information determine the bargaining power of an agent.

As future work, we will extend the present planning model to n -agent systems ($n > 2$). The main challenge of the extension is how to offer side-payment to each other agent in the situation of unknowing other agents' demands (obviously equal distribution does not work). Secondly, it is interesting to extend the current work to nondeterministic cases. This requires to redefine the solution concept and the COACHIEVE algorithm in strong [6] or probabilistic style. Finally, more general mechanisms can be designed for multi-agent planning to deal with changing goals, incomplete information [9,10], and reasoning agents [11,12].

References

1. Ben Larbi, R., Konieczny, S., Marquis, P.: Extending Classical Planning to the Multi-agent Case: A Game-theoretic Approach. In: Mellouli, K. (ed.) ECSQARU 2007. LNCS (LNAI), vol. 4724, pp. 731–742. Springer, Heidelberg (2007)
2. Bowling, M., Jensen, R., Veloso, M.: A Formalization of Equilibria for Multiagent Planning. In: IJCAI 2003, pp. 1460–1462 (2003)
3. Brafman, I.R., Domshlak, C., Engel, Y., Tenenholz, M.: Planning Games. In: AAAI 2009, pp. 73–78 (2009)
4. Brafman, I.R., Domshlak, C.: From One to Many: Planning for Loosely Coupled Multi-agent Systems. In: ICAPS 2008, pp. 28–35 (2008)
5. Brainov, S., Sandholm, T.: Power, Dependence and Stability in Multiagent Plans. In: AAAI 1999, pp. 11–16 (1999)
6. Cimatti, A., Pistore, M., Roveri, M., Traverso, P.: Weak, Strong, and Strong Cyclic Planning via Symbolic Model Checking. *Artificial Intelligence* 147, 35–84 (2003)
7. Moses, Y., Tenenholz, M.: Multi-entity Models. *Machine Intelligence* 14, 63–88 (1995)
8. Muthoo, A.: *Bargaining Theory with Applications*. Cambridge University Press, Cambridge (1999)
9. Wei, H., Zhonghua, W., Yunfei, J., Lihua, W.: Observation Reduction for Strong Plans. In: IJCAI 2007, pp. 1930–1935 (2007)
10. Wei, H., Zhonghua, W., Yunfei, J., Hong, P.: Structured Plans and Observation Reduction for Plans with Contexts. In: IJCAI 2009, pp. 1721–1727 (2009)
11. Tran, S., Sakama, C.: Negotiation Using Logic Programming with Consistency Restoring Rules. In: AAAI 2009, pp. 930–935 (2009)
12. Zhang, D.: Reasoning about bargaining situations. In: AAAI 2007, pp. 154–159 (2007)

Point-Based Bounded Policy Iteration for Decentralized POMDPs

Youngwook Kim¹ and Kee-Eung Kim²

¹ Search Solutions, Seongnam-si, Korea
youngwook.kim@nhn.com

² Korea Advanced Institute of Science and Technology, Daejeon, Korea
kekim@cs.kaist.ac.kr

Abstract. We present a memory-bounded approximate algorithm for solving infinite-horizon decentralized partially observable Markov decision processes (DEC-POMDPs). In particular, we improve upon the bounded policy iteration (BPI) approach, which searches for a locally optimal stochastic finite state controller, by accompanying reachability analysis on controller nodes. As a result, the algorithm has different optimization criteria for the reachable and the unreachable nodes, and it is more effective in the search for an optimal policy. Through experiments on benchmark problems, we show that our algorithm is competitive to the recent nonlinear optimization approach, both in the solution time and the policy quality.

1 Introduction

The decentralized POMDP (DEC-POMDP) is a popular framework for modeling decision making problems where two or more agents have to cooperate in order to maximize a common payoff, and to act based on imperfect state information. While the DEC-POMDP can be applied to many domains such as network routing and multi-robot coordination, it is known to be intractable for computing an optimal policy [1].

In this paper, we are interested in solving infinite-horizon DEC-POMDPs by searching in the space of fixed-size finite state controllers (FSCs). Specifically, we represent the individual policy for each agent as a stochastic FSC in which the nodes correspond to action selection strategies and the transitions correspond to observation strategies. There have been proposed a number of methods for finding FSC policies, but most relevant to our work are the bounded policy iteration for DEC-POMDPs (DEC-BPI) [2] and the nonlinear optimization approach (NLO) [4].

We propose an improved version of DEC-BPI that addresses some of the limitations that prevent the algorithm from finding an FSC policy with a high quality. Our insight for the improvement is based on the observation that we need different optimization criteria depending on whether a controller node in FSC is reachable or not. We show the effectiveness of the proposed algorithm via experiments on standard benchmark problems.

2 Background

A decentralized partially observable Markov decision process (DEC-POMDP) is a multi-agent extension to the POMDP framework. More formally, a DEC-POMDP is defined as tuple $\langle I, S, b_0, \{A_i\}, \{Z_i\}, T, O, R \rangle$ where

- I is a finite set of agents
- S is a finite set of states shared by all agents
- b_0 is the initial state distribution, where $b_0(s)$ denotes the probability that the system starts in state s
- A_i is a finite set of actions available to agent i ; the set of *joint actions* is denoted as $\vec{A} = \prod_{i \in I} A_i$
- Z_i is a finite set of observations available to agent i ; the set of *joint observations* is denoted as $\vec{Z} = \prod_{i \in I} Z_i$
- T is a transition function where $T(s, \vec{a}, s')$ denotes the probability $P(s'|s, \vec{a})$ of changing to state s' from state s by executing joint action \vec{a}
- O is an observation function where $O(s, \vec{a}, \vec{z})$ denotes the probability $P(\vec{z}|\vec{a}, s)$ of making joint observation \vec{z} when taking joint action \vec{a} and arriving in state s .
- R is a reward function where $R(s, \vec{a})$ denotes the shared reward received by all agents when taking joint action \vec{a} in state s .

Since the state is not directly observable and the observations are local to each agent, the agent chooses actions based on its own local histories. This mapping from local observation histories to actions comprises a *local policy*, and the set of every agent's local history comprises a *joint policy*.

A popular representation for policies in infinite-horizon problems is to use *stochastic finite state controllers* (FSCs). The local policy for agent i is represented as a stochastic FSC $\pi_i = \langle Q_i, \psi_i, \eta_i \rangle$, where

- Q_i is the finite set of controller nodes,
- ψ_i is the action selection strategy for each node, where $\psi_i(q, a)$ denotes the probability $P(a|q)$ of choosing action a in node q ,
- η_i is the observation strategy for each node, where $\eta_i(q, a, z, q')$ denotes the probability $P(q'|q, a, z)$ of changing to node q' from node q when executing action a and making observing z .

The set of π_i for each agent i comprises a joint policy $\vec{\pi}$, and the set of nodes from each agent's controller comprises a joint node.

2.1 Bounded Policy Iteration for DEC-POMDPs

Bernstein *et al.* [2]'s bounded policy iteration for DEC-POMDPs (DEC-BPI) is an extension of the bounded policy iteration algorithms for POMDPs [3] to the multi-agent case. It is a greedy local search algorithm that iteratively improves a joint stochastic FSC with a fixed number of nodes by alternating between policy evaluation and improvement. In the policy evaluation step, DEC-BPI computes the value function of the current joint controller by solving the Bellman equation. In the policy improvement step, DEC-BPI randomly selects one of the nodes of an agent, and solves the linear program to obtain an improved controller.

2.2 Nonlinear Optimization Approach

Amato *et al.* [4]’s nonlinear optimization (NLO) takes a more direct approach to obtaining an optimal controller. The problem is formulated as a nonlinear program (NLP) and a state-of-the-art NLP solver is used to find solutions. Since the problem is nonconvex, most of the NLP solvers yield only a locally optimal solution, as in the case with DEC-BPI.

3 Point-Based DEC-BPI

Before we present our algorithm, let us take a closer look at the policy improvement in DEC-BPI. The linear program tries to find better parameters for a node, assuming that we use the controller with the new parameters for the first time step, and then the one with the old parameters from the second step on.

We want the intermediate FSCs during the iterations of DEC-BPI to represent the set of policies that perform well with respect to various reachable state distributions starting from b_0 , but DEC-BPI does not necessarily show this behavior since the monotonic improvement condition requires improving the value for *all* state distributions.

Table 1. Point-based DEC-BPI

```

B ← SAMPLEBELIEFS()
repeat
  Vπ ← EVALUATE(π)
  C ← REACHABLENODESTATES(B, π, Vπ)
  (ε, π̄) ← IMPROVEPOLICY(π, Vπ, C)
until no improvement in any node of any agent
  
```

The main idea behind our point-based DEC-BPI is to have different optimization criteria depending on whether or not a controller node is reachable from the set of useful nodes. The overall algorithm is shown in Table 1, and in the remainder of this section, we explain each step of the algorithm.

3.1 Sampling Beliefs

Since it is intractable to find the exhaustive set of reachable multiagent beliefs under the optimal policy, we approximate the set by sampling from a random policy, similar to [6]. Formally, given a T -step joint tree policy and a joint history $\vec{h}_T = \langle \vec{a}_1, \vec{z}_1, \dots, \vec{a}_T, \vec{z}_T \rangle$ of actions and observations from time step 1 to T , the associated (unnormalized) state distribution $b(\vec{h}_T, \cdot)$ is recursively computed by

$$b(\vec{h}_T, s') = O(s', \vec{a}_T, \vec{z}_T) \sum_s T(s, \vec{a}_T, s') b(\vec{h}_{T-1}, s)$$

where \vec{h}_{T-1} is the sub-history from time step 1 to $T - 1$, and $b(\vec{h}_0, s) = b_0(s)$.

Table 2. Procedure REACHABLENODESTATES

```

C ← {}
for each belief b ∈ B do
  fb ← argmax{fi:Hi→Qi} ∑ $\vec{h},s$  b( $\vec{h},s$ )V $\vec{\pi}$ (f( $\vec{h}$ ),s)
  for each joint history  $\vec{h} \in \vec{H}$  and state s ∈ S do
    if b( $\vec{h},s$ ) > 0 then
      C ← C ∪ {(fb( $\vec{h}$ ),s)}
    end if
  end for
end for
repeat
  for all (q',s') s.t. (q',s) ∈ C and T $\vec{\pi}$ (q',s,q',s') > 0 do
    C ← C ∪ {(q',s')}
  end for
until no more node-state pair to add
    
```

Table 3. The linear program in point-based DEC-BPI for improving reachable node q_i. The variable x(a_i) represents ψ_i(q_i,a_i), and x(a_i,z_i,q'_i) represents η_i(q_i,a_i,z_i,q'_i). P(a_{-i}|q_{-i}) denotes ∏_{k≠i} ψ_k(q_k,a_k), and P(q'_{-i}|q_{-i},a_{-i},z_{-i}) denotes ∏_{k≠i} η_k(q_k,a_k,z_k,q'_k).

```

Variables: ε, x(ai), x(ai,zi,q'i)
Objective: Maximize ε
Improvement constraints: ∀(qi,q-i,s) ∈ C,
  V(q',s) + ε ≤ ∑ $\vec{a}$  P(a-i|q-i)[x(ai)R(s, $\vec{a}$ ) + γ ∑s',z,q':(q',s') ∈ C x(ai,zi,q'i)
  P(q'-i|q-i,a-i,z-i)T(s, $\vec{a}$ ,s')O(s', $\vec{a}$ ,z)V(q',s')]
Unreachability maintenance constraints: ∀(qi,q-i,s) ∈ C and ∀(q',s') ∉ C
  ∑ $\vec{a},\vec{z}$  P(a-i|q-i)x(ai,zi,q'i)P(q'-i|q-i,a-i,z-i)T(s, $\vec{a}$ ,s')O(s', $\vec{a}$ ,z) = 0
Probability constraints:
  ∑ai x(ai) = 1,    ∀ai,zi ∑q'i x(ai,zi,q'i) = x(ai)
  ∀ai x(ai) ≥ 0,    ∀ai,zi,q'i x(ai,zi,q'i) ≥ 0
    
```

3.2 Reachability of Nodes and States

Once we evaluate the value of current policy, we identify the set of useful joint nodes. Formally, a joint node q' of joint controller π is useful for belief b ∈ B if it maximizes the value at the belief. Intuitively, the useful nodes are the candidate initial nodes if the system starts at the state distributions dictated by the belief b. Once we have identified the set of useful joint nodes, we examine the reachability of all the joint nodes from the useful joint nodes. Table 2 shows the overall pseudo-code for finding the set of reachable joint nodes given the set B of beliefs.

Table 4. The linear program in point-based DEC-BPI for improving unreachable node q_i with respect to local history h_i in belief b . $P(s', \vec{z}|s, \vec{a})$ is a shorthand notation for $T(s, \vec{a}, s')O(s', \vec{a}, \vec{z})$.

<p>Variables: $x(a_i), x(a_i, z_i, q'_i)$ Objective: Maximize</p> $\sum_{h_{-i}, s} b(h_i, h_{-i}, s) \sum_{\vec{a}} P(a_{-i} f^b(h_{-i})) \left[x(a_i)R(s, \vec{a}) + \gamma \sum_{s', \vec{z}, \vec{q}': \langle \vec{q}', s' \rangle \in C} x(a_i, z_i, q'_i) P(q'_{-i} f^b(h_{-i}), a_{-i}, z_{-i}) P(s', \vec{z} s, \vec{a}) V(\vec{q}', s') \right]$ <p>Unreachability maintenance constraints: $\forall h_{-i}, s$ with $b(h_i, h_{-i}, s) > 0$ and $\forall \langle \vec{q}', s' \rangle \notin C$</p> $\sum_{\vec{a}, \vec{z}} P(a_{-i} f^b(h_{-i})) x(a_i, z_i, q'_i) P(q'_{-i} f^b(h_{-i}), a_{-i}, z_{-i}) T(s, \vec{a}, s') O(s', \vec{a}, \vec{z}) = 0$ <p>Probability constraints as in Table 3</p>

3.3 Modified Policy Improvement

As in DEC-BPI, our algorithm randomly selects one of the node of an individual controller and uses an LP solver to find new parameter values that improve the controller. The joint node \vec{q} is defined to be reachable if there exists state s such that $\langle \vec{q}, s \rangle \in C$, and the node q_i is defined to be reachable if there exists q_{-i} such that $\vec{q} = \langle q_i, q_{-i} \rangle$ is reachable. If the node q_i selected for improvement is reachable, we solve the LP shown in Table 3. Note that the monotonic improvement is only concerned with reachable joint nodes and states. On the other hand, if the selected node q_i is unreachable, we solve the LP shown in Table 4. The LP essentially tries to make the selected node useful for some belief.

4 Experiments

We implemented all three algorithms discussed in this paper: DEC-BPI, NLO, and point-based DEC-BPI. We used two DEC-POMDP problems for the experiments: decentralized tiger [5] and box-pushing [7]. Our implementation of DEC-BPI was actually the biased version of DEC-BPI, which takes into account the reachability of joint nodes and states by computing the occupancy distribution. We also accordingly modified the LPs for point-based DEC-BPI using the occupancy distribution in order to favor biased policy improvement. The beliefs were collected from randomly instantiated 1-step and 2-step tree policies. The number of beliefs for each problem is: 11 for decentralized tiger and 20 for box-pushing.

We ran each algorithm on each problem, starting from randomly instantiated stochastic controllers with varying number of nodes. We executed 20 runs for each controller size, measuring the value and the wall clock time of each run.

Figure 1 and 2 show the value and time results on the problems. Point-based DEC-BPI was able to yield controllers that attain values much higher than those from DEC-BPI, while taking a fraction of time compared to NLO. Note that in

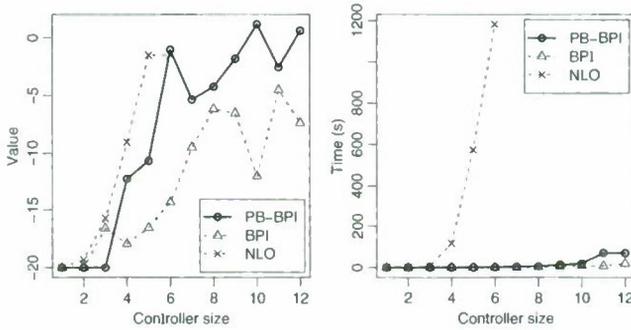


Fig. 1. Value and time results on the decentralized tiger problem

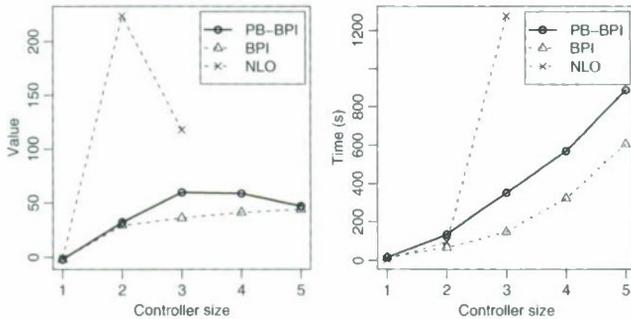


Fig. 2. Value and time results on the box-pushing problem

the case of box-pushing, we believe that we need more nodes than reported here in order to have the performance comparable to NLO, since there are many more reachable beliefs than other problems.

References

1. Bernstein, D.S., Givan, R., Immerman, N., Zilberstein, S.: The complexity of decentralized control of Markov decision processes. *Mathematics of Operations Research* 27(4), 1192 (2002)
2. Bernstein, D.S., Hausen, E.A., Zilberstein, S.: Bounded policy iteration for decentralized POMDPs. In: *Proceedings of IJCAI*, p. 1205 (2005)
3. Poupart, P., Boutilier, C.: Bounded finite state controllers. In: *Proceedings of NIPS*, p. 1209 (2003)
4. Amato, C., Bernstein, D.S., Zilberstein, S.: Optimizing memory-bounded controllers for decentralized POMDPs. In: *Proceedings of UAI*, p. 1241 (2007)
5. Nair, R., Tambe, M., Yokoo, M., Pynadath, D., Marsella, S.: Taming decentralized POMDPs: Towards efficient policy computation for multiagent settings. In: *Proceedings of IJCAI*, p. 1206 (2003)
6. Szer, D., Charpillat, F.: Point-based dynamic programming for DEC-POMDPs. In: *Proceedings of AAAI*, p. 1207 (2006)
7. Seuken, S., Zilberstein, S.: Memory-bounded dynamic programming for DEC-POMDPs. In: *Proceedings of IJCAI*, p. 1208 (2007)

Chinese Named Entity Recognition Based on Hierarchical Hybrid Model

Zhihua Liao¹, Zili Zhang^{2,3}, and Yang Liu⁴

¹ Hunan Normal University, Changsha, Hunan 410081, China

² Southwest University, Chongqing 400715, China

³ Deakin University, Geelong, VIC 3217, Australia

⁴ Jilin University, Changchun, Jilin 130000, China

liao.zhihua61@gmail.com, zzhang@deakin.edu.au, liuyangkok@163.com

Abstract. Chinese named entity recognition is a challenging, difficult, yet important task in natural language processing. This paper presents a novel approach based on a hierarchical hybrid model to recognize Chinese named entities. Three mutually dependent stages – boosting, Markov Logic Networks (MLNs) based recognition, and abbreviation detection – are integrated in the model. AdaBoost algorithm is utilized for fast recognition of simple named entities first. More complex named entities are then piped into MLNs for accurate recognition. In particular, the left boundary recognition of named entities is considered. Lastly, special care is taken for classifying the abbreviated named entities by using the global context information in the same document. Experiments were conducted on People's Daily corpus. The results show that our approach can improve the performance significantly with *precision* of 94.38%, *recall* of 93.89%, and $F_{\beta=1}$ value of 93.97%.

1 Introduction

Named entity recognition (NER) is widely acknowledged as one of the central tasks in natural language processing (NLP). The essential goal of NER is to identify and classify certain proper nouns, such as person names (PER), organizations (ORG), locations (LOC), and so on. NER has attracted much attention in the research community for a long time. Sun et al. proposed a class-based language model to Chinese NER using different models to identify different types of name entities (NEs) in Chinese text[1]. Yu et al. successfully used a high-performance boosting algorithm to handle the Chinese NER task[2]. However, the results remain unsatisfactory.

To this end, a novel approach based on a hierarchical hybrid model is proposed to recognize Chinese NEs. Three mutually dependent stages, namely, boosting, Markov Logic Networks (MLNs) based simple recognition, and abbreviated NEs detection are integrated. The experimental results indicate that the hierarchical hybrid approach can improve the performance significantly. In Section 2, the hierarchical hybrid model is presented. Experiments and results are given in Section 3. Section 4 is concluding remarks.

2 The Hierarchical Hybrid Model

2.1 Simple Named Entities Recognition

Boosting is chosen to recognize the simple NEs based on two reasons. Firstly, a logical semantic and syntactic unit in natural language is the word. The character is a basic written unit in Chinese language and has no real meaning. Consequently, word segmentation is the fundamental task which transforms a Chinese character string into word sequence. Another reason is to gain the high accuracy performance of simple Chinese NEs. Boosting technique in machine learning can meet both requirements in Chinese NER. Since Yu et al. have successfully applied the high-performance boosting technique called AdaBoost.MH to the Chinese NER task [2,3], we use this technique directly.

2.2 Complicated and Compound Named Entities Recognition

While the boosting algorithm can identify many simple NEs, some organizations and locations are difficult to identify due to lack of linguistic knowledge. Take the organization “重庆直辖市政府/Chongqing Municipality Government” as an example, “重庆/Chongqing” is the name part of location, “直辖市/Municipality” is the salient word of location, and “政府/Government” is the salient word of organization; the three parts can conjunct to an integral as a complicated organization. Through the above observations, we incorporate human knowledge via MLNs to validate the boosting NER hypotheses [3]. Since MLNs can easily transform some linguistic knowledge into first-order logic formulas, MLNs were chosen to recognize the complicated and compound NEs [4,5,6].

Now we present an MLN for our task. The main evidence predicate in the MLN is $TaggedEntity(te, i, c)$, which is true iff tagged entity te appears in the i th position of the c th sentence. Punctuation marks are not treated as separate tagged entities; rather, the predicate $HasPunc(c, i)$ is true iff a punctuation mark appears immediately after the i th position of the c th sentence. The predicate $SalientWord(te, sw, i, c)$ is true iff tagged entity te in the i th position of the c th sentence ending with salient word sw , which $sw \in \{LocSalientWord, OrgSalientWord\}$. The query predicates are $InField(f, i, c)$. $InField(f, i, c)$ is true iff the i th position of the c th sentence is part of field f , where $f \in \{Location, Organization\}$, and inferring it performs recognition.

Now we describe our recognition model via MLN. Generally, different types of NEs have different structures [7]. Typical structures of location and organization are as follows:

$$Person \rightarrow \{[last\ name][first\ name]\}(title\ word)$$

$$Location \rightarrow \langle name\ part \rangle * \langle salient\ word \rangle$$

$$Organization \rightarrow \{[person\ name][place\ name][kernel\ name][org.\ name]\} \\ * [org.\ name] \langle salient\ word \rangle$$

Here $\langle \rangle *$ means repeating one or several times. $\{ \} *$ means selecting at least one of items. Meanwhile, since the left boundary of Chinese NEs is more difficult to

recognize than right boundary due to lack of the salient word, both identifiers of $\langle \rangle^*$ and $\{ \}^*$ need to be used to segment effectively the left boundary of Chinese NEs. Therefore, all the above linguistic knowledge can be represented by these rules of such forms as follows:

$$\text{TaggedEntity}(+tc, i, c) \wedge \text{SalientWord}(+te, +sw, i, c) \Rightarrow \text{InField}(+f, i, c)$$

$$\exists te' \text{TaggedEntity}(+tc', i, c) \wedge \text{TaggedEntity}(+te, i + 1, c) \wedge \text{SalientWord}(+te, +sw, i + 1, c) \wedge \neg \text{HasPunc}(e, i) \Rightarrow \text{InField}(+f, i, c)$$

Furthermore, we make the following hard constraint to recognized NEs. Firstly, all kinds of tagged entities are within 25 Chinese characters. Secondly, since all NEs are proper nouns, the tagged entities should end with noun words. Then we define three evidence predicates, which are $\text{NamedEntity}(ne, i, c)$, $\text{LengthEntity}(\text{low_25}, ne, c)$ and $\text{EndWith}(nw, ne, c)$, respectively. $\text{NamedEntity}(ne, i, c)$ is true iff NE ne appears in the i th position of the c th sentence. $\text{LengthEntity}(\text{low_25}, ne, c)$ is true iff NE ne appeared in the c th sentence is lower than or equal to 25 Chinese characters. And the $\text{EndWith}(nw, ne, c)$ is true iff NE ne appeared in the c th sentence ends with noun word. Both constraints are represented by a simple rule:

$$\text{NamedEntity}(ne, i, c) \Rightarrow \text{LengthEntity}(\text{low_25}, ne, c) \wedge \text{EndWith}(nw, ne, c)$$

After constructing these rules, MLNs can learn and perform inference to recognize complicated and compound NEs.

2.3 Named Entity Abbreviation Recognition

After recognizing the complicated or compound Chinese NEs, there may still exist some abbreviated NEs which are difficult to identity. The aim in this stage is to further improve the accuracy by recognizing the abbreviated NEs. In a document, some NEs usually appears in the abbreviated formats in the latter text after firstly appearing in the full format. This enhance the difficulty of recognition. Fortunately, the global feature from the same document may play a key role for recognizing abbreviated NEs. A method to detect abbreviated Chinese NEs can be described as follows: First, constructing a Static Chinese Named Entity List (SCNEL) and recognized NEs are stored in the SCNEL. Second, constructing a Dynamic Candidate Word List (DCWL) and all candidate words are stored in the DCWL. Third, every candidate word in the DCWL has a corresponding feature and initially this feature is set to 0. When candidate word is the random conjunctions of one or more characters of the Chinese NEs in the SCNEL, the feature is set to 1.

3 Experiments and Results

Experiments were conducted using People's Daily Corpus of January 1998 (http://icl.pku.edu.cn/icl_groups/corpus/dwldform1). It is tagged with

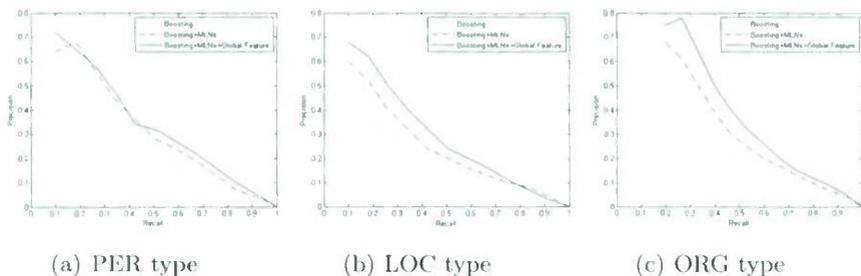


Fig. 1. Precision and recall curves for different NE types

Table 1. Chinese NER performance based on the hierarchical hybrid model in three mutually dependent stages

Approach	NE Type	Precision			Recall			$F_{\beta=1}$		
		All	Left	Right	All	Left	Right	All	Left	Right
Boosting	PER	90.46	87.48	89.67	89.32	87.40	90.49	90.39	87.28	89.51
	LOC	81.25	80.29	81.55	80.94	79.84	80.99	79.15	79.20	80.35
	ORG	79.84	76.86	79.94	66.24	65.34	66.45	72.36	71.34	72.42
	Total	79.84	81.30	83.41	81.75	78.31	80.89	82.87	80.84	82.88
Boosting+MLNs	PER	97.46	96.48	93.67	97.32	97.40	97.49	96.94	96.40	96.81
	LOC	92.86	91.67	92.01	94.37	93.41	93.47	93.75	91.73	92.20
	ORG	90.06	89.78	90.15	89.97	89.79	90.11	90.01	89.73	90.08
	Total	93.15	93.15	93.20	93.24	93.21	93.30	93.06	93.05	93.11
Boosting+MLNs	PER	97.52	97.12	97.13	97.37	97.05	97.09	97.46	97.01	97.03
	LOC	93.21	92.52	92.53	94.53	93.78	94.07	94.16	94.09	94.12
+Global Feature	ORG	90.47	90.41	90.43	90.79	90.70	90.73	90.67	90.66	90.69
	Total	94.38	94.32	94.39	93.89	93.76	93.79	93.97	93.49	93.56

Note: "All" represents both of boundaries, "Left" left one and "Right" right one.

POS according to Chinese Text POS Tag Set. The Corpus contains 26 million characters. The first half of the corpus is used as the training corpus, and testing corpus is corresponding to the later half month of January 1998, respectively. The parameters in the experimental evaluation include precision, recall and $F_{\beta=1}$.

We first used the decision stump as the weak classifier in boosting to recognize the simple NEs. And we ran the AdaBoost.MH for 5000 rounds. Then we took the learning and inference algorithms provided in the open-source Alchemy package to recognize the complicated and compound NEs[5]. We performed discriminative weight learning using the voted perception algorithm, and inference using the MC-SAT algorithm. In MLN weight learning, we used 100 iterations of gradient descent and chose the default values except it. The total learning time reached to 24 hours. In inference, we ran MC-SAT for 20 hours. Finally, we constructed a SCNEL and a DCWL to help recognize the abbreviated NEs and used java language to program a matching recognition code. The experimental platform is on a server with two CPUs at 2.8 GHz and 4GB of memory. The experimental results are shown in Figure 1. From these figures, we can conclude that the results using the combination of boosting and MLNs are clearly more

accurate than those of the boosting method, and MLNs significantly improve the performance of accuracy. Furthermore, although our hierarchical hybrid method trivially surpasses the combination of boosting and MLNs for PER, our method perform well in LOC and ORG.

As shown in Table 1, the hierarchical hybrid model achieves satisfactory improvements with *precision* of 94.38%, *recall* of 93.89%, and $F_{\beta=1}$ value of 93.97%. Besides, the difference between left boundary and right one falls from 2% in the first stage to 0.5% in the third stage. It means that through linguistic knowledge MLNs perform well in left boundary recognition.

4 Conclusion

A novel approach based on hierarchical hybrid model was proposed to recognize Chinese NEs. This model incorporates three mutually dependent stages into a unifying framework. Experiments were conducted on People's Daily corpus. The results show that our approach can significantly improve the performance and achieves a fairly satisfactory result. Future work is to extend this approach to larger datasets.

References

1. Sun, J., Gao, J., Zhang, L., Zhou, M., Huang, C.: Chinese named entity identification using class-based language model. In: ICL, pp. 1–7 (2002)
2. Yu, X., Carpuat, M., Wu, D.: Boosting for chinese named entity recognition. In: The 5th SIGILLAN Workshop on Chinese Language Processing, pp. 150–153 (2006)
3. Yu, X.: Chinese named entity recognition with cascaded hybrid model. In: NAACL HLT 2007, pp. 197–200 (2007)
4. Richardson, M., Domingos, P.: Markov logic networks. *Machine Learning* 62, 107–136 (2006)
5. Kok, S., Singla, P., Richardson, M., Domingos, P., Summer, M., Poon, H.: The alchemy system for statistical relational ai (2006)
6. Singla, P., Domingos, P.: Discriminative training of markov logic networks. In: AAAI 2005, pp. 120–128 (2005)
7. Wu, Y., Zhao, J., Xu, B.: Chinese named entity recognition combining a statistical model with human knowledge. In: The ACL Workshop on Multilingual and Mixed-Language Named Entity Recognition, pp. 65–72 (2003)

Text Disambiguation Using Support Vector Machine: An Initial Study

Doan Nguyen¹ and Du Zhang²

¹ Hewlett-Packard Company
8000 Foothills Blv
Roseville, California 95747
doan.nguyen@hp.com

² Department of Computer Science
California State University
Sacramento, California 95819-6021
zhangd@ecs.csus.edu

Abstract. Word segmentation is an essential step in building natural language applications such as machine translation, text summarization, and cross-lingual information retrieval. For certain oriental languages where word boundary is not clearly defined, a recognition process can become very challenging. One of the serious problems is dealing with word ambiguity. In this paper, we investigate the use of Linear Support Vector Machines (LSVM) for word boundary disambiguation. We empirically show, in the Vietnamese case, that LSVM obtains a better result when comparing to the Trigram Language Model approach.

Keywords: Word Segmentation, Ambiguity Resolution, Covering Ambiguity Resolution, Trigram Language Model, and Linear Support Vector Machines.

1 Introduction

In Oriental languages, there are no explicit word separators such as space as in English to indicate word boundaries. Word segmentation is a process of dividing written text into meaningful units, such as words. There are two common sub-problems with word segmentation: (1) out-of-vocabulary (OOV) words identification and (2) ambiguity resolution.

In a sequence of Vietnamese syllables, *S*, composing of two syllables *A* and *B* occurring next to one another, if *S*, *A*, and *B* are words each, then there is a covering ambiguity in *S*. For example, the two syllables string “*Nhật ký*” can be interpreted as three different words in the two sentences below.

Sentence 1: *Nhật* (Japan) | *ký* (signs) | *hiệp định* (agreement) | *về* (about) | *giảm* (reduce) | *khí thải* (gas emission) | *nhà kính* (greenhouse).

Sentence 2: *Nhật ký* (diary) | *đời* (life) | *sinh viên* (student).

From the given example, we observe that making a right word choice is not a trivial task for a computer. This determination has to be obtained from knowing the context of a sentence where a word is used.

2 Related Works

For Vietnamese language, researches in word ambiguity resolution are still at an early stage. The work of Le et al [3] attended to overlapping ambiguity except the covering ambiguity problem. This work has an impressive precision and recall rates at 95% and 96.3% respectively. Nguyen [2,4], in WebSBA, employed Web data for word segmentation. This work addressed ambiguities resolution using bigram language model and word collocation concepts. Its result was compared against another popular Vietnamese Word Segmentation approach - the JVNsegmenter [5]. There was no discussion on how ambiguities are handled in [5]. WebSBA had a precision and recall rates at 89% and 82%.

The Chinese word segmentation has a similar covering ambiguity problem. Xiao et al [6] regarded the covering ambiguity problem to word sense disambiguation. They used vector space model to formulate the contexts of ambiguous words. For 90 frequent words, the authors manually trained 77,654 sentences. They obtained a 96.58% accuracy rating. Recently, Su-qin Feng [1] collected contextual information statistics of covering ambiguous words and found a context calculation mode by using log likelihood ratio. Fourteen frequently appeared covering ambiguous words are used for evaluation. The highest evaluation accuracy rate reaches 95.60%.

3 Proposed Approaches

A segmenter produces a segmentation result. Because of using a Longest Matching strategy [2,4] in favoring compound words, a covering ambiguity error might still exist within the segmented text. We examine segmented disyllabic words in a sentence, in a left to right order, for a potential ambiguous word. If a word matches a rule description in section 3.1, we formulate a text chunk consists of this word and its context words in its neighborhood (about ± 4 words). This text is evaluated using a disambiguating module detailed in sections 3.2 or 3.3 below.

3.1 Patterns for Ambiguity Detection

The main difference between our work and the previous works is to process unknown ambiguous words and to resolve them. We defined the following patterns, denoted in BNF, which could contribute to ambiguity checking process:

- **<syllable-preposition>** ::= <syllable> <cho (for) | ở (at) | trên (above) | với (with, to) | trong (inside) | ... >. For example, the word “*trắng trong*”.
- **<noun-pronoun-verbs>** ::= <noun | pronoun> < là (to be) | làm (to do) | đi (to go) | cần (to need) | ... >. For example, the word “*người làm*”.
- **<syllable-pronoun>** ::= <syllable> < Selected Pronouns: first, second person, or kinship terms: *tôi* (I) | *anh* (you) | ... >. For example, the word “*đàn anh*”.
- **<Irregularity_in_word_shape>** ::= <first_letter_lower_case(syllable)> <first_letter_upper_case(syllable)>. For example, the word “*bà Ba*”.

3.2 Trigram Language Model

A language model is usually formulated as a probability distribution $p(s)$ over a string s that attempts to reflect how frequently a string s occurs as a sentence in a corpus [7].

We estimate the trigram probabilities using a large corpus of text using their trigram frequencies formulated in equation 1:

$$f_3(w_3 | w_1, w_2) = \frac{C(w_1 w_2 w_3)}{C(w_1 w_2)} \quad (1)$$

where C is the count of sequences $w_1 w_2 w_3$ and $w_1 w_2$ appearing in a corpus. The trigram probabilities in a corpus can be estimated linearly as follows:

$$p(w_3 | w_1, w_2) = \alpha_3 p_3(w_3 | w_1, w_2) + \alpha_2 p_2(w_3 | w_2) + \alpha_1 p_1(w_3) \quad (2)$$

where α is a tuning parameter, or a weight, with $\alpha \in [0, 1]$. We find $0 < \alpha_3, \alpha_2, \alpha_1 < 1$ by optimizing on "held-out" data. A string, with a higher probability score, is expected to contain a corrected segmented word(s).

3.3 Linear Support Vector Machines (LSVMs), Text Representation and Feature Selection

Many problems in natural language processing can be categorized as classification problem. Covering ambiguity resolution can be regarded in the same fashion. That is if an ambiguous word should be divided into two individual words, a separated condition (-1) or combined as a single word, a combined condition (+1). We will use LSVM [10] to determine it.

Each chunk of text (a text chunk), consists of an ambiguous word and other context words nearby (± 4 words), and is represented as a vector of words. For text classification simpler binary feature values (i.e. a word either occurs, a 1 value, or does not occur, a 0 value) are often used instead [10]. We also eliminate noise words such as "và" (and), "của" (of), "khi" (when), etc. These words are not essential to an overall context of a text chunk.

3.4 Learning Support Vector Machines

We are motivated by a method called active learning [9] in suggesting which text from the pool to use in the learning process. This feature enables a reduction in utilizing human resources for training of examples. Our algorithm of a pool-based active learning is described as follows:

```

Algorithm ActiveLearner
Inputs:
  Pool of unlabeled examples; Initial Classifier;
Output:
  Updated classifier;
begin
  (1) Classify all unlabeled examples;
  (2) Separate examples into 2 partitions where
      each example's decision score has a closest
      distance to its partition's centroid value;
  (3) Trainer trains the classifier with new
      labeled examples in a partition which has a
      higher centroid value;
end.
```

In the aforementioned algorithm, the pool of unlabeled examples consists of text chunks collected from the Web via a Yahoo! web service. Collected examples are classified to obtain their decision values using SVM^{light} [8]. We then separate these examples into two partitions where each contains examples having decision values closest to its centroid. A centroid value of a partition is obtained by taking average of the decision values of its examples. Finally, a trainer trains only examples belonging to an upper partition which has a higher centroid value. Figure 1 shows two scatter plots. Plot 1 contains all unlabeled examples in a pool intended for training. Plot 2 contains a much reduced pool of unlabeled examples suggested by the system for a trainer to train.

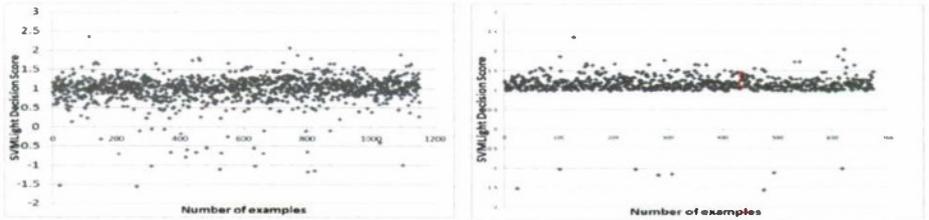


Fig. 1. Scatter Plot 1 (left) contains all examples requested for training. Scatter Plot 2 (right) contains actual selected examples to be trained by a trainer.

Heuristically, we choose to select examples located in an upper partition, examples shown in Plot 2, since these examples are expected to change the maximum margin hyperplane the most [9].

3.5 Classification by Support Vector Machines

Once a covering ambiguous word is detected by a word segmentation algorithm, a text chunk is formulated. This text chunk contains the ambiguous word itself and neighboring words in about ± 4 words. Two possible interpretations of this word, combining or splitting form, are the possible outcomes of the SVM^{light} 's classifier.

Here is an example of queried vectors:

0 57022:1 67833:1 70589:1 75251:1 # *nhật* | *ký* | *đời* | *sinh viên*
 0 44403:1 67833:1 75251:1 # *nhật ký* | *đời* | *sinh viên*

The SVM^{light} returns two real values, of a decision function, for each examined vector respectively:

0.99976741
 0.99981328

We implement the following decision table in finalizing a decision given decision values from the SVM^{light} .

	Decision Values from <i>SVM^{light}</i>			
Vector combined word form	Positive Value, $f(x)$	Negative Value, $f(x)$	Positive Value, $f(x)$	Negative Value, $f(x)$
Vector separated words form	Positive Value, $f(y)$	Negative Value, $f(y)$	Negative Value, $f(y)$	Positive Value, $f(y)$
Final Decision	Combine word.	Separate words.	If $f(x) > f(y) \Rightarrow$ Combine word; Separate words otherwise.	Take the combined form (default behavior).

4 Experiments

We used WebBSA [2] as our tested segmenter. We evaluated three methods: (1) Longest Match Rule; (2) Trigram language model; and (3) LSVM. We started with a raw text corpus of 166,484 text titles to estimate trigram frequencies. The data is serving for our corpus need in building trigram frequencies and training of its model. From the same list, we randomly took 10,300 document text titles, for ease of data extraction, and performed word segmentation, with method (1). We also located ambiguous words using word patterns (section 3.1). We identified and learned 1,535 texts having about 120 potential covering ambiguous words included. This low number could be a reflection as observed from [3]. Using 120 ambiguous words, we fetched the Web to obtain another 1,153 texts. With active learning, only 675 texts (about 58%) were selected for the training. From the same pool, we also randomly selected another set of texts for non-active learning. We learned these examples in addition to the 1,535 examples trained earlier. In our final task, we fetched the Web with another unknown 3,174 texts. These texts are serving as our unseen test set. We identified the result as follows:

Table 1. Evaluations with test set.

Evaluation Category	Accuracy
Rule-based approach (Longest Matching Rule)	80.4%
Trigram Language Model	77.2%
LSVM with learning of random examples	88.8%
LSVM with active learning	92.5%

The above result indicates that the SVM with active learning approach outperforms all the other approaches with unknown test set data.

5 Conclusion

Two possible approaches to disambiguate covering word ambiguity have been described for languages where word boundary is not clearly defined. Our test result

confirms that the Learning-based approach has advantages in term of flexibility, better accuracy and scalability. For the Vietnamese word segmentation works we studied [2,3,4,5], we believe that this is a first attempt to address the covering ambiguity condition specifically. For the future work, we plan to increase the scope of experiment to increase larger volume of held out data for testing. We are also looking into a possibility to integrate this concept to address the overlapping ambiguity condition.

References

1. Feng, S.-q., Hou, S.-q.: Context-Based Approach for Covering Ambiguity Resolution in Chinese Word Segmentation. In: 2009 Second International Conference on Information and Computing Science, ICIC, vol. 2, pp. 43–46 (2009)
2. Nguyen, D.: Using Search Engine to Construct a Scalable Corpus for Vietnamese Lexical Development for Word Segmentation. In: The 7th Workshop on Asian Language Resources (ALR7). Conjunction with ACL-IJCNLP 2009, Suntec City, Singapore (2009)
3. Lê, H.P., Nguyen, T.M.H., Roussanaly, A., Ho, T.V.: A hybrid approach to word segmentation of Vietnamese texts. In: 2nd International Conference on Language and Automata Theory and Applications, Tarragona, Spain (2008)
4. Nguyen, D.: Query preprocessing: improving web search through a Vietnamese word tokenization approach. In: SIGIR 2008, pp. 765–766 (2008)
5. Nguyen, C.T., Nguyen, T.K., Phan, X.H., Nguyen, L.M., Ha, Q.T.: Vietnamese word segmentation with CRFs and SVMs: An investigation. In: Proceedings of the 20th Pacific Asia Conference on Language, Information and Computation (PACLIC 2006), Wuhan, CH (2006)
6. Luo, X., Sun, M., Tsou, B.K.: Covering ambiguity resolution in Chinese word segmentation based on contextual information. In: Proceedings of the 19th International Conference on Computational Linguistics, Taipei, Taiwan, August 24-September 1, pp. 1–7 (2002)
7. Chen, S.F., Goodman, J.: An empirical study of smoothing techniques for language modeling. Center Research in Computing Technology, Harvard University, TR-10-98 (1998)
8. Joachims, T.: Making large-Scale SVM Learning Practical. In: Schölkopf, B., Burges, C., Smola, A. (eds.) *Advances in Kernel Methods - Support Vector Learning*. MIT Press, Cambridge (1999)
9. Tong, S., Koller, D.: Support vector machine active learning with application to text classification. In: Proceedings of the Seventeenth International Conference on Machine Learning (2000)
10. Dumais, S.T., Platt, J., Heckerman, D., Sahami, M.: Inductive learning algorithms and representations for text categorization. In: Proceedings of ACM-CIKM 1998, pp. 148–155 (November 1998)

Diacritics Restoration in Vietnamese: Letter Based vs. Syllable Based Model

Kiem-Hieu Nguyen and Cheol-Young Ock

Natural Language Processing Lab, School of Computer Engineering and Information
Technology, University of Ulsan, Korea
nguyenkiemhieu@gmail.com, okcy@ulsan.ac.kr

Abstract. In this paper, we present some approaches to diacritics restoration in Vietnamese, based on letters and syllables. Experiments with language-specified feature selection are conducted to evaluate contribution of different types of feature. Experimental results reveal that combination of Adaboost and C4.5, using letter-based feature set, achieves 94.7% accuracy, which is competitive with other systems for diacritics restoration in Vietnamese. Test data for diacritics restoration task in Vietnamese could be freely collected with simple preprocessing, whereas large test data for many natural language processing tasks in Vietnamese is lack. So, diacritic restoration could be used as an application-driven evaluation framework for lexical disambiguation tasks.

Keywords: Lexical disambiguation, diacritics restoration, decision tree, boosting, word segmentation, feature space.

1 Introduction

The aim of diacritics restoration is to restore original script from diacritic-free script by correct insertion of diacritics. Subjects of diacritics restoration are languages containing diacritics, such as French, Spanish, Dutch, Vietnamese, etc. In natural language processing, diacritics restoration is a particular lexical disambiguation task. For example, “xu ly ngon ngu tu nhien”, which means “natural language processing”, which is a diacritic-free script in Vietnamese, will be restored as “xử lý ngôn ngữ tự nhiên” after inserting correct diacritics.

Though potential commercial applications (typing assistant, search query resolution, etc.) could be found from diacritics restoration, not many researchers have studied on this topic. A well-known work in the literature is accents restoration in Spanish and French [1]. Considering accents restoration as multinomial classification, Yarowsky achieved accuracy of 99.7% on the full task using decision list learning. In a comparative work, Mihalcea compared learning from letters and learning from words for diacritics restoration in Romanian [2]. Conclusion from this paper is that learning from letters, which is comparable with learning from words, could be applied to resources-scarce languages. In an attempt to use dictionary in combination with learning from words, accuracy over 90% was achieved in [6]. It's worth to notice that training data and test data for

supervised learning of diacritics restoration are easily collected from Internet. It is opposite to other lexical disambiguation tasks, where annotation of training data and test data is the most time consuming phase.

To our knowledge, grapheme based and word based methods in [5] and constrained sequence classification based method in [7] are the only researches on diacritics restoration in Vietnamese. In [5], accuracy of memory-based classifiers reaches 63.1% and 82.7% with learning from words and learning from graphemes in that order. In that paper, lexical diffusion, which is the division of tokens containing diacritics and all tokens, is claimed to measure the difficulty of diacritics restoration task in a specific languages. According to this measurement, Vietnamese is the second (after Yoruba) in 14 studied languages.

In this paper, experiments on diacritics restoration in Vietnamese are conducted using five strategies: learning from letters, learning from semi-syllables, learning from syllables, learning from words, and learning from bi-grams. To focusing on comparing proposed approaches, we only use C4.5 as classifier. On the other hand, to focus on performance, AdaBoost and C4.5 are combined to get high accuracy.

The paper is organized into four parts. The first part briefly introduces current researches related to diacritics restoration. The second part describes in more detail linguistic characteristics of Vietnamese to figure out difficulties of diacritics restoration in Vietnamese in comparison with other languages. The next part describes the feature set in five proposed approaches. In the last part, experimental results are showed and are discussed to point our advantages and disadvantages of proposed approaches. The paper ends with remarked conclusions and future works.

2 Fundamental Lexical Units in Vietnamese

Vietnamese alphabet contains 29 letters, including 12 vowels and 17 consonants. English letters like [f, j, w, z] are not included, where as [ã, â], [ê], [ô, ơ], [ê], [ư] and [d] are variants of [a], [e], [o], [u], [d] in that order. In writing language and speaking language, tone marks are added to vowels to adjust different tones.

Diacritics restoration in Vietnamese must resolve two kinds of ambiguity: phonetic diacritics ambiguity (e.g. between [a], [ã], and [â]) and tonic accents ambiguity (e.g. between [a], [á], [à], [ã], [â], and [ạ]). Considering diacritics restoration as multinomial classification, combination of phonetic diacritics and tonic accents ambiguities is one of the reasons that makes the task in Vietnamese more difficult than in other languages.

Vietnamese is a monosyllable language. For example, in the phrase “xử lý ngôn ngữ tự nhiên” (natural language processing), tokens which can be separated by space are syllables. Raw text in Vietnamese does not contain explicit words boundary. Word segmentation is the task of defining this boundary. For example, above phrase, “xử lý ngôn ngữ tự nhiên”, as input of a word segmentation system should have output as “[xử lý] [ngôn ngữ] [tự nhiên]”, where words boundaries are explicit. Sequence of syllables in each pair of brackets indicates a word. A

appear in the corpus as tokens. Remaining 15.5K tokens don't belong to Vietnamese dictionary, each of which rarely appears in the corpus. They are mainly English named entities, like celebrities' names, movie titles, song titles, country names, locations, terminologies, etc., all of which don't contain diacritics. Some tokens containing diacritics are acronyms, noisy or misspelling text. To eliminate effect of noisy data and to reduce feature space in decision tree learning, all tokens not belonging to Vietnamese dictionary (out-of-vocabulary tokens) are tagged with the same label "UNKNOWN".

3 Feature Set

In our work, surrounding context of the ambiguous pattern is selected as features. A sliding window scanned through training corpus to build data instances. Following popular experiments in the literature of lexical disambiguation, we chose the window of size 5 to the left and to the right of the ambiguous pattern. The ambiguous pattern is centered on the sliding window. No feature selection or parameter tuning is applied. Default parameters of C.45 implemented in Weka are used.

Table 3. Statistic of number of syllables in words in a Vietnamese dictionary containing 30k entries

#Syllable in a word	#Words	Percentage
1	5208	17.27
2	22866	75.81
3	1362	4.52
4	653	2.16
≥ 5	75	0.25

Five feature types are used:

1. *Learning from letters*: Ambiguous patterns are letters that may have different diacritics (Table 1). Attribute values are case sensitive. Delimiters (space, comma, dot, question mark, and colon), date, and numbers are tagged as SPACE, COMMA, DOT, QUESTION, COLON, DATE, and NUMBER, respectively.
2. *Learning from syllables*: In 20K unique tokens in the corpus, 15.5K tokens are out-of-vocabulary tokens, where no diacritics restoration needs to be applied. To reduce feature space of training data, all out-of-vocabulary tokens are tagged with the same label "UNKNOWN". 4.5K tokens which are syllables used in Vietnamese dictionary, have equivalent 1.3K diacritic-free tokens after removing diacritics.
3. *Learning from semi-syllables*: Focusing on reduction of feature space, we propose an approach based on construction rules of syllables in Vietnamese, called learning from semi-syllables. In learning from syllables, each attribute has 1.3K values. Semi-syllables are extracted by omitting head consonants from syllables. As the result, each attribute has about 100 values.

4. *Learning from words*: To prepare data for learning from words, training text is preprocessed by word segmenter. In our work, we use word segmenter¹ in [4] which is claimed to produce 90% accuracy.
5. *Learning from bi-grams*: To clarify difference between learning from syllables (unigrams) and learning from words, learning from n-grams is considered as an “intermediate approach”. In Vietnamese dictionary, majority of words are composed of 2 syllables (Table 3). As a result, bi-gram based learning was chosen in our work.

4 Experimental Results and Future Works

Using training corpus, 2M data instances of all ambiguous patterns are created in each learning approach. The evaluation follows 10-fold cross validation schema. The highest accuracy is achieved by combining C4.5 as the weak learner and AdaBoost as the boosting learner. AdaBoost improves the accuracy 1.4% against individual C4.5.

Table 4. Comparison of accuracy in different learning strategies

Learning strategy	Accuracy
Baseline (most frequent class)	45.15
C4.5 + Letters	93
C4.5 + Semi-syllable	88.2
C4.5 + Word	91.9
C4.5 + Bi-gram	88.8
AdaBoost + C4.5 + Letters	94.7

Despite of the simplicity of features set, learning from letters results in the best performance. Learning from semi-syllables produces, as expected, lowest accuracy. Although the lost of information is obvious when all head consonants are omitted, a 1.6% penalty against learning from syllables for the reduction of feature space (from 1000 to 100 candidates for each feature) is an encouraged result.

Discussion about using n-gram model or using word segmentation as preprocessing phase in mono-syllables languages like CJK or Vietnamese is continuing while high accuracy in word segmentation have not been achieved [8]. In our work, learning from words performs better than learning from syllables and learning from bi-grams. It is our belief that more accurate word segmenter will improve the results of learning from words. Using word based diacritics restoration as an application-driven evaluation framework for word segmentation task is a potential future work.

Constrained sequence classification based method in [7] achieves 94.3% accuracy, which is in line with our best result. It should be noticed that training data and test data in our work and in [7] are different. Experiments using the same training data and test data should be conducted to get reliable comparison.

¹ <http://www.loria.fr/~lehong/tools/vnTokenizer.php>

5 Conclusions

In this paper, experiments on diacritics restoration are conducted using different learning strategies. Experiments results reveal that learning from letters achieves the best result. On the other hand, performance of other strategies is expected to be improved by using accurate syntactic and semantic knowledge extracted from raw text. Our proposed strategy, learning from semi-syllables, produces slightly lower results than other strategies. However, reasonable dimensionality of feature space and potential improvement of accuracy shows that learning from semi-syllables is not a bad choice. In the worst case, it could be used as a baseline method.

References

1. Yarowsky, D.: Decision lists for lexical ambiguity resolution: application to accent restoration in Spanish and French. In: Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics, pp. 88–95 (1994)
2. Mihalcea, R.F.: Diacritics Restoration: Learning from Letters versus Learning from Words. In: Gelbukh, A. (ed.) CILing 2002. LNCS, vol. 2276, pp. 96–113. Springer, Heidelberg (2002)
3. Mitchell, T.M.: Decision Tree Learning. *Machine Learning*, 52–78 (1997)
4. Hông Phnong, L., Thi Minh Huyền, N., Roussanaly, A., Vinh, H.T.: A hybrid approach to word segmentation of Vietnamese texts. In: Martín-Vide, C., Otto, F., Fernau, H. (eds.) LATA 2008. LNCS, vol. 5196, pp. 240–249. Springer, Heidelberg (2008)
5. De Pauw, G., et al.: Automatic Diacritic Restoration for Resource-Scarce Languages. In: Proceedings of 10th International Conference of Text, Speech and Dialogue, Pilsen, Czech Republic, September 3–7 (2007)
6. Simard, M.: Automatic Insertion of Accents in French Text. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, EMNLP-3, Granada, Spain (1998)
7. Truyen, T.T., et al.: Constrained Sequence Classification for Lexical Disambiguation. In: Ito, T.-B., Zhou, Z.-H. (eds.) PRICAI 2008. LNCS (LNAI), vol. 5351, pp. 430–441. Springer, Heidelberg (2008)
8. Nie, J.Y., et al.: On the Use of Words and N-grams for Chinese Information Retrieval. In: Proceedings of the Fifth International Workshop on Information Retrieval with Asian Languages, Hong Kong, China, September 30–October 1 (2000)

Tag Quality Feedback: A Framework for Quantitative and Qualitative Feedback on Tags of Social Web

Tae-Gil Noh¹, Jae-Kul Lee¹, Seong-Bae Park^{1,*}, Seyoung Park¹, Sang-Jo Lee¹, and Kweon-Yang Kim²

¹ Department of Computer Engineering
Kyungpook National University
702-701 Daegu, Korea

{tgnoh, jklee, sbpark, sypark, sjlee}@sejong.knu.ac.kr

² School of Computer Engineering, Kyungil University,
Gyeongsan 712-701, Korea
{kykim}@kiu.ac.kr

Abstract. A feedback framework is proposed in this paper to assist Web 2.0 users' taggings. A new measure called *Estimated Daily Visit* is defined and proposed as the measure for tag quality. Quantitative and qualitative feedback methods are also defined with the measure. A prototype has been implemented to show the validity of the framework, and preliminary result shows that the framework can successfully enhance quality of tags on user-generated contents.

1 Introduction

Folksonomies, tag annotations of user-generated contents, are now become a common standard for Social Web services. Images or videos in top search results are often well annotated with multiple tags of fine granularity. This can lead to false impression that user-generated contents are now well annotated with tags. However, this is not true. Contents that are annotated inadequately simply do not exposed in the search result, due to their poor quality of annotations. Many contents are annotated with no tags, too few or too general tags that cannot help search engines to find the content.

Open nature of tag annotation is a sword of two edges: Creative users can always add new but useful tags that are helpful to describe and differentiate their content. Yet naive users often annotate their contents with tag words that will never be used as a search term, or sometimes, they don't even bother to tag at all.

The goal of this research is to provide help for this second group of users. The paper proposes an interactive framework that helps non-expert users to understand the *quality* of their tags at the tagging time. The proposed framework can provide information about the tag words being attached to content interactively:

* Corresponding author.

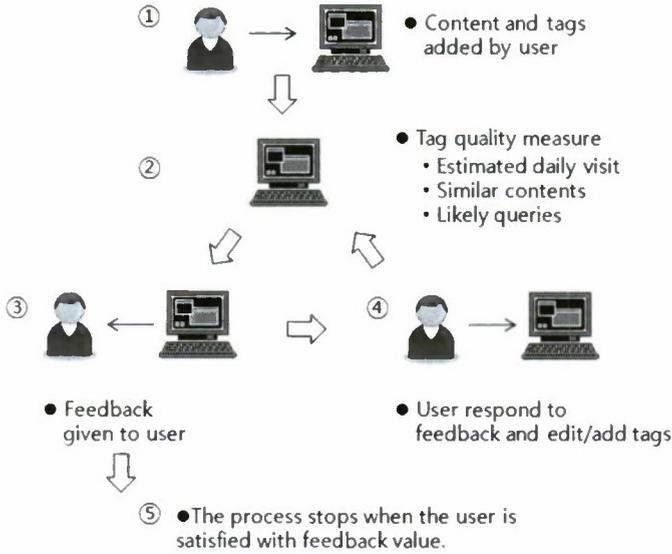


Fig. 1. Process of tag quality feedback from user's point of view

- How useful are current tags as search keywords?
- How specific is the tag set? Should the user need to add more terms?
- What other contents are there with similar/same tag set? And how many?
- Does the tag set make annotated content distinctive enough?

In this paper, a set of measures for tag quality is first proposed. The measures are then used in a framework of interactive feedback designed to help Web service users. We call this framework as *tag quality feedback*. A prototype implementation of this framework has been done to show the validity of the framework, and preliminary result shows that the framework can successfully enhance quality of tags on user-generated contents.

2 Related Work and Basic Idea of the Framework

Assisting users at the tagging time is not a new idea. Tag recommendation is one of such scheme [1][2]. Also there are commercial tools that help finding tags, useful links and related pictures for the user-generated content [3][4]. The proposed framework of this paper has a very different focus compared to previous work of tag recommendations. Focus of this research is on comparing tags being attached on the content and tags previously attached on existing contents.

The idea of assigning "quality value" for annotated tags appears in previous work like [5]. However, previous quality values for tags are generated by reliability of authors, or redundancy of tags annotated on the same contents.

Figure 1 shows the tag quality feedback process from the user's point of view. A user first adds a content and its initial tags. The tags are then analyzed by

the framework in terms of search and retrieval. The quantitative and qualitative feedbacks are then given to the user, and the user refines her tags. Then, the cycle starts again. It stops when the user is satisfied at the estimated quality. The framework helps users to decide how much tag is enough, and to see where the attached tag set will put the content among related contents.

3 Tag Quality Measures

3.1 Estimated Daily Visit

A good annotation should not only correctly reflect the content (relation between content-to-annotation), but also should perform well as an index that makes the content distinctive (relation between annotation-to-annotation and annotation-to-query). As an index, the role of annotation is to help other users to locate the content. Estimated daily visit count (EDV) is proposed as a measure of tags in this role. Let Q be a set of queries where each member q_i is a query (with one or more terms) which will make a search result that includes the current content being annotated. Then EDV can be formulated as follows:

$$EDV = tdv \times \sum_{q_i} P_q(q_i) \times P_s(q_i) \quad (1)$$

In the equation, tdv is total daily visit count for the whole service, P_q is the probability of the query q_i to be presented as a query, and $P_s(q_i)$ is the probability of the content in focus to be visited in the search result of query q_i .

For example, if a picture is tagged with "Eiffel Tower" and "Paris", three queries can reveal the content in their search result. $Q = \{ \text{Eiffel Tower, Paris, Eiffel Tower AND Paris} \}$. EDV for this content is determined by sum of three values: $P_q(\text{Eiffel Tower})P_s(\text{Eiffel Tower}) + P_q(\text{Paris})P_s(\text{Paris}) + P_q(\text{Eiffel Tower AND Paris})P_s(\text{Eiffel Tower AND Paris})$.

In the EDV equation, the summed up probability is then multiplied by total daily visit. As a result, EDV value will show "how often your content will be visited by users via a search engine, with current set of tag words". Actual calculation of P_q and P_s is depending on the implementation.

In general, EDV prefers tag sets with following conditions:

- Larger tag set than smaller tag set: a tag set with more tags has more ways (queries) to access the content. In the equation, set Q becomes larger with more tags.
- Commonly used terms than unknown/rare words: P_q part goes near zero if the term is not often used in queries.
- Distinctive combinations than common combinations: This is due to P_s . Distinctive combinations of common query words will yield higher EDV value.

3.2 Associated Measures for Interactive Comments

To help users' understanding, three sub-measures are proposed in this paper. These values are flag values (boolean values) that can be feedback to the user to notify possible problems of the current annotation.

- Too few tags: if the number of set Q is smaller than a given threshold, or average of both average P_q and P_s is smaller than a given threshold, this flag value will be set. This flag value can only be set for a tag set, not for each individual tag.
- Terms too rare/unknown: if average P_q value is lower than a given threshold, this flag value can be set. This value can be set for each tag and a tag set.
- Too indistinctive combinations of tags: if average P_s value is smaller than a given threshold. This flag value will be set. This value can only be set for a tag set.

Flag values will be shown to users in UI as comments for tags.

4 Framework for Quantitative and Qualitative Tag Quality Feedback

4.1 Qualitative Feedback

Quantitative measures are often not the best method for human users to see the "position" of their annotation. To help users to visualize the effect of their tags, the proposed framework additionally has two qualitative feedback methods.

Listing of similar contents. Especially for tag annotations for image or moving pictures, this is an effective feedback for users to understand where the annotated tags will put their content among other contents. By comparing the tag set attached on the content in focus with other tag sets, it is possible to show some random contents that are tagged with similar tags. By showing top n similar contents, this method can achieve its goal of letting user to know what other contents are similar in terms of annotated tags.

Listing of likely queries. Reaching the content can be done by more than one set of queries, thus this method can be regarded as a method to show likely paths that will lead other users to the content.

4.2 Architecture of Tag Quality Feedback Framework

With EDV and two qualitative feedback methods, it is possible to draw the architecture of tag quality feedback framework.

Figure 2 shows overview of the framework. Three major modules of the framework lie on the right side of the figure: P_q , P_s and qualitative feedback module. They gain the data needed to calculate feedback values from query log data and the search engine on the right. From these major modules, three quantitative values and two qualitative feedbacks are generated. The generated values are then passed to UI for each cycle of tag quality feedback.

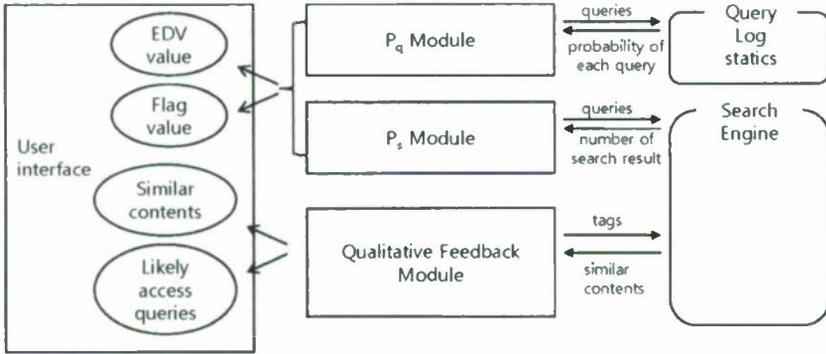


Fig. 2. Overview of tag quality feedback framework

4.3 Prototype Implementation and Preliminary Result

A prototype system has been implemented to test the feasibility of tag quality feedback framework. The prototype is implemented as a local program that is designed to improve qualities of tags annotated on pictures. The program assumes that a new picture is being uploaded for Flickr. The search results and statistic values for tags are gained by Flickr APIs.

Several model probabilities and constant values must be set before implementations. tdv value is a constant that represent the total number of visits on the contents of the services. In the prototype, it was set as 100 million. $P_q(q_i)$ is a value that represents probability of the query q_i to be submitted as a query. Modeling probability of query q_i is an interesting issue. In our prototype, it was not possible to access the query logs of the target service, and P_q has been replaced by probability of a term to be appeared as a tag. That is, for q_i with single term, $P_q(q_i) = \text{number of term observed} / \text{number of all tags observed}$. Also for p_i with more than one term, it was defined as $P_q(q_h \text{ AND } q_j) = P_q(q_h)P_q(q_j)$. $P_s(q_i)$ is a value that represents probability of the content in focus to be visited among the search result of query q_i . In the prototype simple IDF-like value was used. That is, $P_s(q_i) = c / \text{number of contents in the search result of } q_i$. Constant c is the average number of visiting upon a search result. In the prototype, optimistic value of 40 has been used as constant c .

The qualitative feedback was also prepared similarity. To get similar contents, members of set Q are queried upon the target service sequentially from q_i with the longest one to the smallest one. The first 20 pictures gained by this method are shown to the user as similar contents. Listing of likely queries can be gained by providing a number of top q_i with higher $P_q(q_i)P_s(q_i)$. In the prototype, three most likely queries and the actual number of contents resulting from each query are given back to users.

With this setup, a small preliminary experiment was done with 100 selected Flickr images. The images have been selected from larger set of images that have only one tag with minimum EDV value of 1.0. Two test users were requested to

use the prototype system to interact and add tag annotations to each picture. Testers have been instructed to interact with the feedback output at least two cycles. The refined tags after the feedback achieved much higher quality in EDV values: average EDV value of 42.1 and 57.3. The average number of tags were 4.3 and 7.2.

This preliminary experiment shows that the proposed tag quality feedback framework can enhance quality of tags annotated on user-generated contents. However, this preliminary experiment cannot replace real accessibility test of the system. For example, it is not shown yet how normal users would accept the response/feedback, or how this feedback would change typical behavior of naive users. Also, it is yet to be shown that EDV value and the actual number of content visit have positive correlations. Evaluating various aspects of tags, tag quality feedback and Web 2.0 users is prominent future work for the tag quality feedback framework.

5 Conclusions

A framework for tag quality feedback is proposed in this paper. A measure called “estimated daily visit” is first derived to reflect the likelihood of annotated tags to reveal the content in terms of keyword search. Three associated measures and two qualitative feedback methods are also devised to help naive users to edit their tags to get better score. There is a lot of future work remains for the framework. Assumptions of EDV are based on search processes, and to prove those assumptions are right, tag sets optimized to EDV should actually have significant higher number of content visitors. It would need a long time evaluation with real-world environment, and it would also need controlled contents with original tag annotations and EDV optimized tag contents.

Acknowledgements. This work was supported by the Industrial Strategic Technology Development Program (10035348, Development of a Cognitive Planning and Learning Model for Mobile Platforms) funded by the Ministry of Knowledge Economy (MKE, Korea).

References

1. Jschke, R., Marinho, L., Hotho, A., Schmidt-Thieme, L., Stumme, G.: Tag Recommendations in Folksonomies. In: Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery (2007)
2. Sigurbjornsson, B., Van Zwol, R.: Flickr tag recommendation based on collective knowledge. In: Proceedings of the 17th International Conference on World Wide Web (2008)
3. Zemanta, <http://www.zemanta.com>
4. Opencalais, <http://www.opencalais.com>
5. Xu, Z., Fu, Y., Mao, J., Su, D.: Towards the semantic web: Collaborative tag suggestions. In: Proceedings of Collaborative Web Tagging Workshop at 15th International World Wide Web Conference (2006)

Semantic Networks of Mobile Life-Log for Associative Search Based on Activity Theory

Keunhyun Oh and Sung-Bae Cho

Department of Computer Science, Yonsei University
Sinchon-Dong, Seodaemun-gu, Seoul 120-749, Republic of Korea
ocworld@sclab.yonsei.ac.kr, sbcho@cs.yonsei.ac.kr

Abstract. Recently, due to proliferation of mobile devices, we can collect users' life-log. Human long-term memory is an interconnected network. The retrieval system of it is cue-dependent. Semantic networks are used to implement it of human retrieval system. It is possible to retrieve relevant data more effectively by using a search system based on network visualization which provides relations among data rather than a text-based search system. This paper proposes representation of semantic networks of mobile life-log based on activity theory, and associatively finds data based on network visualization for it. We have implemented the system, searched data from an example of search, and performed a subjective test. As a result, we have confirmed that this system is useful for associative retrieval resembled to human cue-dependent recall.

Keywords: Mobile Log, Semantic Networks, Associative Search, Network Visualization.

1 Introduction

Recently, because of widespread of mobile devices, it is possible to collect and manage various user information through them called mobile life-logs such as a user's calls, SMS (short message service), photography, music-playing and GPS (global positioning system) information. Since the amount of these data increase exponentially, it is important to retrieve data needed.

Semantic networks have a merit for storing mobile life-log. Mobile life-log is one of the auxiliary memory units for a person. Information is saved as an interconnected network in Human long-term memory. Associative search means the cue-dependent retrieval system of human interconnected memory [1]. A representation of mobile life-log should support associative search like human retrieval system. Semantic networks are more suitable than relational database systems for it. In this paper, to make an effective representation of mobile life-logs that express user's context, context model of activity theory is adopted. According to this theory, user's context should be formed by activity [2].

Associative search system is effective to relevant search. In previous studies, text-based associative search systems are mainly presented. It is not enough to fully utilize the strength of semantic networks because it does not include relations between data.

This paper proposes the semantic network representation of mobile life-logs based on activity theory for a visualization-based associative search of human memory.

2 Activity Theory

Activity theory is a powerful and clarifying descriptive tool rather than a strongly predictive theory. The object of activity theory is to understand the unity of consciousness and activity. Activity theory incorporates strong notions of intentionality, history, mediation, collaboration and development in constructing consciousness [2]. Context model of activity theory assumes a subjective view on situations. This is in contrast to the prevailing view where context normally describes an objective defined situation. Any experience is personal [3]. In this paper, a semantic network representation of mobile life-log is designed based on context model of activity theory.

Table 1. Elements of Context model

Type of context	Meaning
Environmental context	Users' surroundings accessed by the user.
Personal context	The mental and physical information about the user.
Social context	The social aspects of the user like roles.
Task context	The user's goals, tasks and activities.
Spatio-temporal context	Time, location and the community present.

3 Proposed Method

After collecting log data which are GPS, Call, SMS, picture viewer, photo, MP3, charging, and action, the system generates a semantic network from mobile life-logs following the defined a representation. It visualizes pre-structured a semantic network. Next, a user search data using selection and keyword associative search on a visualized semantic network. It provides a visualized result graph structured in relational data and relationship among data. The semantic abstraction helps a user understand the result retrieved information. Figure 1. shows the entire system for the mobile life-log semantic network in this paper.

3.1 Design for a Representation of Semantic Networks of Mobile Life-Log

Context model of activity theory is referenced to define a semantic network representation of mobile life-log. In this representation, 'user' node which expresses users' profile is the root node. A next type of node linked up with the root node is 'category' of actions. 'Action' is followed by a category node. Since 'place' and 'date and time' are important factors to infer an action, they are connected to an action node. The last type is related to 'functions' of a mobile device. Table 3 shows the definition of types of nodes. Table 2 provides information on how context model is adapted to this representation.

Table 2. Mapping node types to Elements of context model of activity theory

Context of activity theory	Node Type
Activity	Category, Action
Personal context	User
Task context	Function(Playing a music, taking a picture)
Spatio-Temporal context	Place, Date and Time
Environmental context	Content
Social context	Function (Call, SMS)

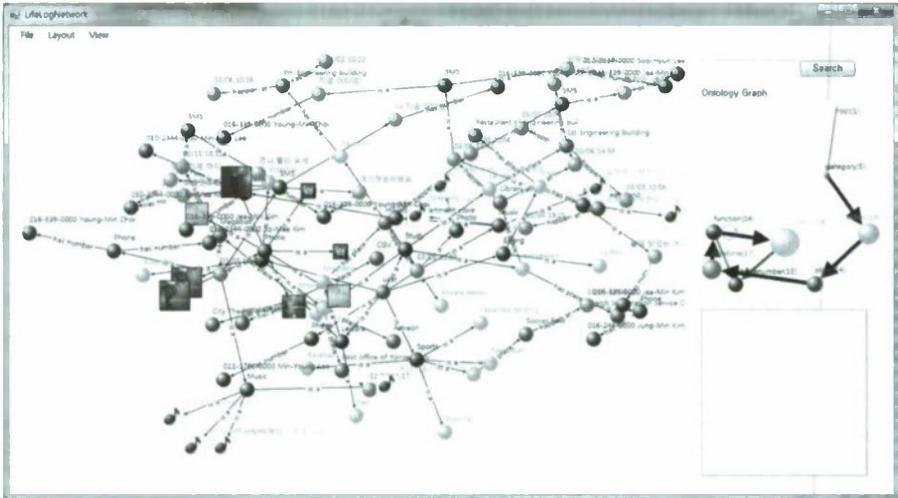


Fig. 1. The associative search system for a mobile life-log semantic network. A user can retrieve data by selection associative search on a left side and by keyword associative search on a right upper side text box and a button. Ontology graph is shown in the center of a right side.

3.2 Associative Search

Text-based associative search systems are limited to associatively search data, since it is hard to express the relationship between data. Therefore, methods of associative search based on network visualization are needed. Selection search means a way to find relative data through selecting a node on semantic networks visualized. When a node is chosen, its directly relative data are shown to a user. Also, a user can click a node, one of the retrieved nodes. A user finds data through selecting a node, step by step. This process is resembled to human retrieving memory.

In addition, this system contains a function of keyword associative search. If the system has an only selection search of finding data, time is spent on retrieving information. The pseudo-code for keyword associative search is introduced by Figure 2. Its input is a keyword as a query, output is a result graph structured in relative precedent nodes, descendent nodes, and their relationship.

<pre> Input: string keyword, Graph S function Graph KeywordAssociativeSearch Graph ResultGraph; Node KeywordNode = DFS(keyword,S).Result(); ResultGraph.add(DFS(keyword,S).Routes()); ResultGraph.add(DFS(Keyword,S).Result()); ResultGraph.add(Traverse(keywordNode,S).Routes()); return ResultGraph; end </pre>
--

Fig. 2. The pseudo-code for keyword associative search

3.3 Semantic Abstract for Semantic Networks

Semantic abstraction can show a representation of a network more effectively. Semantic abstraction introduced by Shen et al. (2006) [4] is adapted to semantic networks in this paper. Ontology graph means a graph of that nodes represent types of nodes of a network and of that edges and their relationship. Semantic abstraction can simplify networks without removing nodes of types they want to find. A simplified network is named an induced graph. An induced graph can be constructed by user's selecting in the ontology graph. Types of nodes are named type nodes and Types of edges means type edge. An instance of Ontology graph is shown in Figure 2.

$$R_i = C * \frac{NC_i}{NC_{max}} \quad (1) \quad E_i = C * \frac{EC_i}{EC_{max}} \quad (2)$$

R_i means a radius of i type node. Maximum size of a type node is named C , a constant value. NC_i is defined as the number of nodes of i type. Also, NC_{max} is the maximum count of nodes of type. E_i refers to width of i type edge. C represents maximum size of a type edge. EC_i is count of edges of i type. EC_{max} is largest in the number of edges of type.

4 Experiments and Evaluation

We use Mobile log data collected from a college student during 3 days. The constructed mobile life-log semantic network contains 109 nodes and 106 edges. Node XL library is used for graph visualization (<http://www.codeplex.com/NodeXL>).

A given query is "What is the message, the SMS, during watching a movie". It means that she does not know any information except for 'SMS' and 'watching a movie'. Figure 3 shows process to traverse a mobile life-log semantic network by using associative search. Selection associative search is shown in Figure 3(a) and (b) Although not enough information is given, a user can find data by reminding relevant data step by step. By using keyword associative search, she can see background of each message Figure 3(c)). If the result graph is very complex, it can be simplified by semantic abstraction (Figure 3(d)).

In order to validate the usefulness of the proposed method, we performed a subjectivity test about the implemented application for ten users based on the System Usability Scale (SUS) questionnaires. The SUS is a simple, ten-item scale giving a global view of subjective assessments of usability where its score has a range of zero to one hundred [5]. Figure 4(a) shows the SUS test results.

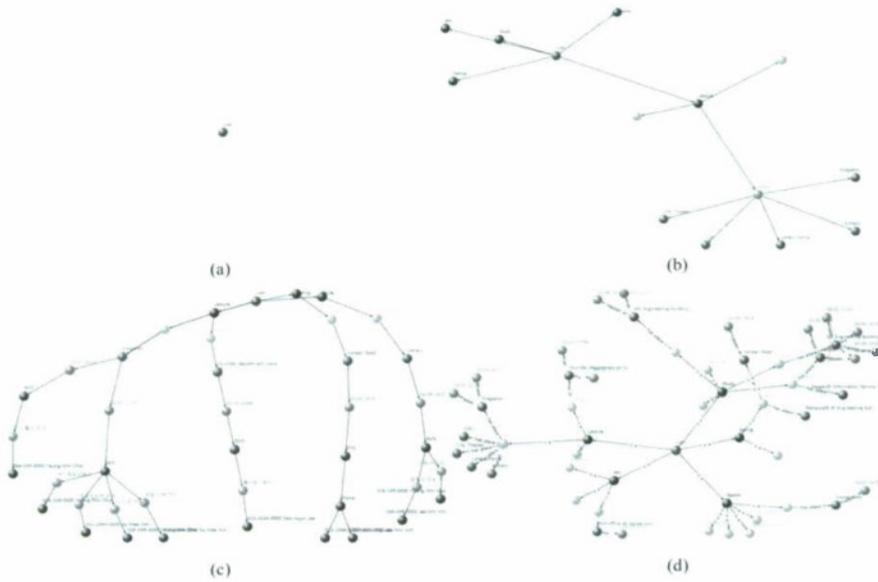


Fig. 3. An example of associative search. (a) the initial state of selection associative search (b) the second state after selecting 'leisure' category and 'watching a movie' action in selection associative search (c)The result graph for the 'SMS' keyword (d) The induced graph extracted from the mobile life-log semantic network except for 'Contents' type, 'Phone number' type, and 'function' type.

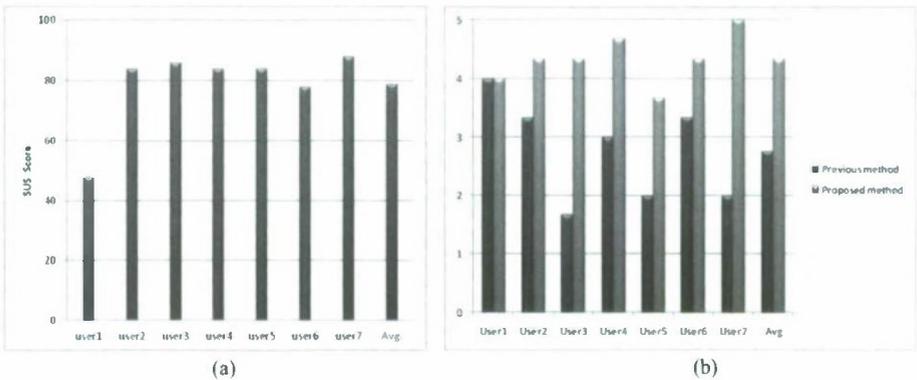


Fig. 4. (a)SUS scores for the proposed system (b) Scores for usability test to evaluate search system

To compare proposed associative search system with previous text-based semantic network method, three questions on Table 3 are added on usability test. This test result is shown as Figure 4(b). These results indicated that the associative search based on visualization provides effective ways to retrieving data.

Table 3. Questionnaires of the usability test to evaluate search system

No.	Questionnaire	Strongly Disagree			Strongly agree	
1	I think search features provided is useful	1	2	3	4	5
2	I think a way to provide search results is effective	1	2	3	4	5
3	Search results is satisfied with me	1	2	3	4	5

5 Conclusion

In this paper, we presented a design for semantic networks of mobile life-log for associative search. Mobile life-logs can support human memory. For human-like retrieval, we stored life-logs in semantic networks. The semantic network representation of mobile life-logs is based on activity theory. In addition, we presented associative search for efficient retrieval based on visualization. It has selection and keyword associative search of that result is shown as a visualized result graph to provide relationship between data. For users' understanding, this graph can be simplified by semantic abstraction. We showed that the proposed method was able to find related data easier and to help users' understanding.

Acknowledgement. This research was supported by the Conversing Research Center Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2009-0093676).

References

1. Raaijmakers, J.G., Shiffrin, R.M.: Search of associative memory. *Psychological Review* 88(2), 93–134 (1981)
2. Nardi, B.A.: Context and consciousness: activity theory and human-computer interaction. The MIT Press, Cambridge (1996)
3. Kofod-Petersen, A., Cassens, J.: Using activity theory to model context awareness. In: Roth-Berghofer, T.R., Schulz, S., Leake, D.B. (eds.) *MRC 2005. LNCS (LNAI)*, vol. 3946, pp. 1–17. Springer, Heidelberg (2006)
4. Shen, Z., et al.: Visual analysis of large heterogeneous social networks by semantic and structural abstraction. *IEEE Trans. on Visualization and Computer Graphics* 12(6), 1427–1439 (2006)
5. Brooke, J.: SUS: A Quick and Dirty Usability Scale. In: Jordan, P.W., et al. (eds.) *Usability Evaluation in Industry*. Taylor and Francis, London (1996)

Three-Subagent Adapting Architecture for Fighting Videogames

Simón E. Ortiz B.¹, Koichi Moriyama², Ken-ichi Fukui²,
Satoshi Kurihara², and Masayuki Numao²

¹ Graduate School of Information Science and Technology, Osaka University

² Institute of Scientific and Industrial Research, Osaka University

8-1, Mihogaoka, Ibaraki, Osaka, 567-0047, Japan

{ortiz,koichi,fukui,kurihara,numao}@ai.sanken.osaka-u.ac.jp

Abstract. In standard fighting videogames, since opponents controlled by computers are in a rut, the user has learned their behaviors after long play and gets bored. Thus we propose an adapting opponent with three subagent architecture that adapts to the level of the user by reinforcement learning. The opponent was evaluated by human users by comparing it against static opponents.

1 Introduction

Fighting videogames are a popular genre of videogames. A fighting videogame is a simulation of hand-to-hand combat and is designed to be played by at least two users competitively. However, these games can also be played by only one user. In this case, the machine will take control of the opponent. If given the option, however, users may prefer to play against other users.

We assume that one of the main reasons users prefer to play against other users is that the AI found in standard videogames is uninteresting. Typically it is of a simple design [1], e.g., Finite State Machines [3], which means that AI in standard videogames is not complex enough to learn users' patterns.

Nevertheless, learning the user behavior and adapting to it in order to *defeat* the user should *not* be the aim. An opponent that behaves so would learn to easily defeat the user and it is not interesting. Therefore, our aim is to *adapt to the level of the user*. Here lies the novelty of our research.

2 Fighting Videogame

In typical fighting videogames the first player that lowers the health-points (HP) of the opponent to zero is the winner of a round. The winner of a fight is the best of several rounds. Some videogames have a time limit per round.

The set of available actions in a game is $X \cup D \cup C \cup B \cup M$. X is the set of *simple attacks*. Simple attacks deal moderate damage. D is the set of defensive actions, or *blocks*. Blocks guard the character from simple attacks. C is the set of *combos*. Combos are predefined combinations of simple attacks

that deal significant damage. B is the set of *combo-breakers*. Combo-breakers are special combinations that counter-attack combos. Each combo $c_i \in C$ might have a different combo-breaker $b_i \in B$. When the corresponding combo-breaker is executed *before* the attacking player finishes delivering the combo, the receiving player will not receive the extra-damage of the combo. M is the set of movements the players use to navigate the character. Different fighting videogames vary in the details and design of the possible actions.

3 Proposal

We propose an agent that learns to adapt to the user in a fighting videogame. This agent controls a character as the opponent of the user. We divided the agent into three subagents, each of which is in charge of handling some types of actions of fighting videogames. They are Main Subagent (MSA), Executing-Combo Subagent (ECSA) and Receiving-Combo Subagent (RCSA). Since videogames must run in real-time, all the learning is delayed until the end of each round. From the agent's point of view, one round equals one *episode*.

3.1 Main Subagent (MSA)

MSA is in charge of executing simple attacks, blocks, and moving. When deemed appropriate, MSA passes the control to one of the other subagents. MSA is modeled as a Profit-Sharing agent [2].

At the t -th turn of the episode n , MSA first recognizes the environment as a state s_t and looks up recorded weights $w_{n-1}(s_t, a^i)$ of all actions a^i available in s_t . After that, MSA chooses an action a_t with the probability calculated from the weights using Boltzmann equation [5], with *temperature* τ :

$$P_{s_t}(a^i) = \frac{\exp(w_{n-1}(s_t, a^i)/\tau)}{\sum_k \exp(w_{n-1}(s_t, a^k)/\tau)}. \quad (1)$$

The agent records the pair (s_t, a_t) and executes a_t . The available actions are those defined in the videogame in question, plus passing control of the character to ECSA or RCSA. After the action has been executed, or the subagent executes its action, MSA resumes control of the character.

At the end of the episode, MSA receives a reward R_n from the environment and updates the weight of all recorded pairs (s_t, a_t) of this episode by the following rule. T is the last turn in the episode.

$$w_n(s_t, a_t) := w_{n-1}(s_t, a_t) + R_n \cdot \gamma^{T-t} \quad (1 \leq t \leq T). \quad (2)$$

MSA receives higher positive rewards when the difference of the final HPs of the agent and the user is small, although negative rewards are given when the difference is significant. This reinforces actions that lead the agent to behave in such a way that it is not too difficult nor too easy for the user. That is, we are reinforcing actions that put the agent at the same level of the user.

3.2 Executing-Combo Subagent (ECSA)

ECSA has the responsibility of choosing combos and executing them. In order for the agent as a whole to be at the same level of the user, the combos the agent executes must also be on a level close to that of the user.

Since the agent must act in real-time during a round, ECSA randomly selects a combo from its combo set $C_A \subseteq C$ and executes the selected one when invoked. Therefore, the problem is how to create C_A . If we consider the set of combos used by the user, $C_U \subseteq C$, the goal of ECSA is to create C_A of similar difficulty.

To create C_A , we need metrics to order sets by their difficulty. We use the following three: *ratio of used combos*, *indistinguishability of combos*, and *entropy of combo-breakers*. In the following definitions of metrics, C is the set of available combos for the game in question, and $C' \subseteq C$ is a set of combos.

Ratio of used combos: A better user would execute a wider variety of combos because it would make it difficult for the opponent to predict the combo-breakers. Hence, the ratio of used combos is a valid metric:

$$\text{used-ratio}(C') = \frac{|C'|}{|C|}. \quad (3)$$

Indistinguishability of combos: Since the combo-breaker must be executed *before* the last action of the combo, a set of combos that are indistinguishable given the initial actions is more difficult than a set where the combos can be distinguished by their initial actions. This can be formalized as follows:

$$\text{inds}(C') = \frac{\sum |\text{combos with repeated initial actions in } C'|}{|C'|}. \quad (4)$$

Entropy of combo-breakers: The set of combos sharing a combo-breaker is easier than that of combos having different combo-breakers, because a player playing against the former need not decide which combo-breaker should be executed. Hence, the entropy of the set of *distinct* combo-breakers, B' , of C' is a valid metric:

$$\text{entr}(C') = \frac{-\sum_{b \in B'} P(b) \log P(b)}{\log |B'|} \quad (5)$$

where $P(b)$ is the probability of randomly choosing a combo-breaker b out of the combo-breakers of C' .

ECSA first creates a combo set containing m combos, whose combo-breakers and initial actions are different (high combo-breaker entropy, low indistinguishability). We consider that such an initial set is not too difficult, but it is not too easy either.

After finishing an episode, this combo set is partially adapted to that of the user by the algorithm presented in Fig. 1. Δ defines the level of tolerance in the difference of sizes of the sets, and `max_iter` limits the number of tries. Although only the subroutine `delete` is presented along with the algorithm, the subroutines `add` and `swap` follow the same idea as `delete`.

```

adapteECSA():
  if (used-ratio(C_A) > used-ratio(C_U) + Δ) delete(C_A);
  elif (used-ratio(C_A) < used-ratio(C_U) - Δ) add(C_A);
  else swap(C_A);

delete(C_A):
  for (i:=0; i < max_iter; i++)
    c:= a combo in C_A chosen randomly;
    if (|entr(C_A\{c})-entr(C_U)| < |entr(C_A)-entr(C_U)|
        or |inds(C_A\{c})-inds(C_U)| < |inds(C_A)-inds(C_U)|)
      C_A := C_A\{c}; return;
  endif
endfor
    
```

Fig. 1. Pseudo-code of ECSA

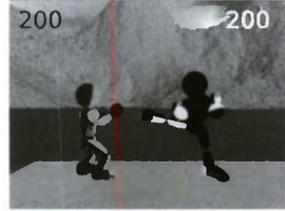


Fig. 2. Fighting videogame

3.3 Receiving-Combo Subagent (RCSA)

The design of RCSA is presented extensively in [4]. This subagent basically mines the patterns of the combos executed by the user after each episode.

When invoked during a round, RCSA matches the combos executed by the user with mined patterns; using the matched patterns, RCSA predicts the next possible combos. Then RCSA chooses the combo-breaker stochastically based on the relative frequency of the predicted combos. For more details, see [4].

4 Experiments

We developed a simple fighting videogame using Crystal Space 3D [6] to test the proposed adapting agent. An image of the videogame is Fig. 2.

The fighting videogame has the following characteristics: the fights occur in a 2D plane; the characters have a height of 3.5 units and a width of 2 units; the stage is a finite platform with a length of 42 units, falling from the platform equals losing the fight; there is no time limit; there is one round per fight; the initial HP of the characters is 200; the set X of simple attacks contains punch (p), kick (k), and special attack (s), the last one being a long range projectile attack; each of the simple attacks deal one point of damage; the set D of defenses contains one action: block; while blocking, simple attacks do not have effect; the set C of combos is listed in Table 1; for a combo to be valid, each action must be executed within 0.5 seconds of the previous one; the combo-breaker of a combo

Table 1. Combos

ID	Act	Damage	ID	Act	Damage
0	pppp	15	6	kppk	20
1	pppk	20	7	kpps	25
2	ppps	25	8	kspp	15
3	pkpp	20	9	kspk	20
4	pkpk	15	10	kpsp	30
5	pkps	25	11	kkkk	30

Table 2. Rewards

HP diff	Reward	HP diff	Reward
< 25	+1.00	< 125	-0.25
< 50	+0.75	< 150	-0.50
< 75	+0.25	< 175	-0.75
< 100	-0.10	≥ 175	-1.00

is defined as its *last action*; if the combo-breaker is valid, the character executing the combo receives its damage; the set M of movements contains move to the right, move to the left, jump and crouch.

The proposed agent was used with the following parameters:

States: To keep the design of the agent simple, we discretized the world state as follows. These were selected because they provided enough information to the agent to make intelligent decisions: (a) crouching or not (agent/user), (b) jumping or not (agent/user), (c) receiving a combo or not (agent), (d) executing a simple attack or not (user), (e) blocking or not (agent/user), (f) at an edge of the platform or not (agent), (g) $HP < 30$ or not (user), (h) the distance between the user and the agent, discretized in eight sections: ≤ 0.25 , ≤ 0.50 , ≤ 2.00 , ≤ 2.60 , ≤ 4.00 , ≤ 10.00 , ≤ 24.50 and > 24.50 , (i) the distance from the agent to the closest special attack thrown by the user, discretized in three sections: ≤ 0.50 , ≤ 2.60 and > 2.60 , and (j) the difference in HP between the user and the agent, rounded to tens.

Actions: The available actions were those available in the game, plus ECSA, RCSA, and stay. Instead of the actions right and left, the agent used approach and withdraw. Approach, withdraw, crouch, and block were executed for 0.1 seconds. Stay had a duration of 0.4 seconds. All the other actions lasted as long as it took to fully execute them.

Rewards: The reward was defined as Table 2. The HP difference was the absolute difference between the HP of the user and the agent.

Others: γ was fixed at 0.99. τ was fixed at 1.0. $m = 3$. $\Delta = 0.1$. `max_iter` = 20. The number of tracked patterns in RCSA [4] was five.

For comparison purposes we developed three static agents: **weak**, **medium** and **strong**. The **weak** agent was very easy to defeat; 50% of its action were to stay; it only executed combos 0 and 10 of Table 1; the combo-breaker was always p. The **medium** agent was obtained by training our adapting agent against a user for 20 rounds in advance, while it did not adapt during fights. The **strong** agent was very difficult to defeat; it always got close to the user and executed one of all available combos randomly whenever close enough; the combo-breakers were chosen stochastically based on the distribution of combo-breakers for the initial actions executed by the user.

We compared these static agents against two versions of our adapting agent: **adap0** and **adapF**. The **adap0** agent was as explained in Section 3. The **adapF** agent was structurally the same as **adap0**, but it had been trained by playing 20 rounds beforehand.

We asked 28 real users to play the game between 15 and 30 rounds against each agent. The users were of different nationalities, ages and with different level of expertise at playing videogames. After the first 15 rounds with an agent, the user could quit whenever he/she was no longer having fun. The users did not know the characteristics of each agent. The order of the agents was randomized for the users. The users filled in a questionnaire after the experiments. They were asked to order the agents from most fun to least fun.

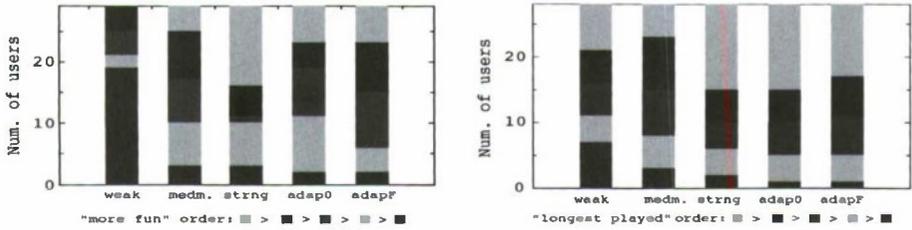


Fig. 3. (Left) Questionnaire results. (Right) Analysis of playing length

The results of the questionnaire are shown in Fig. 3 (Left). This band chart indicates how many subjects rated the opponents as the corresponding rank. The *adapF* received the least amount of negative ratings.

We also compare the length of play against each opponent. For each subject, the opponents were ranked in descending order of the length. In case of draw, both opponents were in the same rank. The comparison is shown in Fig. 3 (Right). Similar to the questionnaire, the proposed agents were the opponents that figured less in the least played opponents.

5 Conclusion

An agent that adapts to the level of the user in a fighting videogame was developed. The adapting agent is divided into three subagents: MSA, ECSA and RCSA, each of which is in charge of handling different aspects of fighting. In comparison with static agents, the adapting agent received the least amount of negative ratings.

References

1. Adams, E.: *Fundamentals of Game Design*, 2nd edn. New Riders, Berkeley (2009)
2. Arai, S., Sycara, K.: Effective learning approach for planning and scheduling in multi-agent domain. In: *Proc. of the 6th International Conference on Simulation of Adaptive Behavior*, pp. 507–516 (2000)
3. Graepel, T., Herbrich, R., Gold, J.: *Learning to fight*. In: *Proc. of the International Conference on Computer Games: Artificial Intelligence, Design and Education*, pp. 193–200 (2004)
4. Ortiz, S., Moriyama, K., Matsumoto, M., Fukui, K., Kurihara, S., Nunao, M.: Road to an interesting opponent: An agent that predicts the users combination attacks in a fighting videogame. In: *Proc. of the Human-Agent Interaction Symposium* (2009)
5. Sutton, R.S., Barto, A.G.: *Reinforcement Learning: An Introduction*. MIT Press, Cambridge (1998)
6. <http://www.crystalspace3d.org/>

Incremental Learning via Exceptions for Agents and Humans: Evaluating KR Comprehensibility and Usability

Debbie Richards and Meredith Taylor

Macquarie University
Computing Department, Faculty of Science
North Ryde, NSW, 2109, Australia
{richards,mtaylor}@science.mq.edu.au

Abstract. Acquiring knowledge directly from the domain expert requires a knowledge representation and specification method that is comprehensible and feasible for the holder and creator of that knowledge. The technique, known as multiple classification ripple down rules (MCRDR), is novelly applied to the problem of building and maintaining a library of training scenarios for use by customs and immigration officer trainees in our agent-based virtual environment which may be indexed for retrieval based on the rules associated with them. Our evaluation study aims to demonstrate the utility of the MCRDR combined case and exception structure rule-based approach over standard rules alone and a non-case-based approach.

Keywords: Ripple down rules, scenarios, training simulation.

1 Introduction

The comprehensibility and usability of knowledge structures have received less attention than their correctness, completeness and consistency [7]. In recognition that acquiring knowledge has been a bottleneck in the development of knowledge based systems (KBS) [5] further leading to validation and maintenance issues, it is important that the knowledge representation and acquisition method be accessible and manageable by a human. In cases where the knowledge is acquired via machine learning, maintenance and acquisition by the human is less of an issue. However, the output of these algorithms should be comprehensible to the human. Quinlan [6] refers to Donald Michie's requirement that concept expressions must be "correct and effectively computable descriptions that can be assimilated and used by a human being" going so far as to regard knowledge representations which are not comprehensible to the domain expert as not qualifying as knowledge.

We are currently developing an agent-based virtual training environment, known as BOrder Security System (BOSS), for trainee airport customs and immigration officers to determine if a passenger should be allowed entry into Australia. Ripple Multiple Classification RDR (RDR) [3] have been used to address many different problems within many application domains. However, novelly we have employed MCRDR to represent and capture the knowledge needed in an agent-based virtual

environment training simulation. The agent’s knowledge (i.e. what to say, what to do, how to respond) and the domain knowledge to be passed to the trainee are both captured using MCRDR. In this way we concurrently and interactively within the training environment train the software agents and the human. The significance of using a KR which can be employed by the domain trainer/expert is that it becomes feasible to deploy the system because we can move beyond a research prototype containing a handful of handcrafted scenarios.

In a previous study involving 36 participants we found a statistically highly significant difference (1.63384E-11) using a one-tailed t-Test: Two-Sample Assuming Equal Variances, in the scores achieved on pre and post test knowledge tests for the border security domain after using our system. Given that our training system was found to be a useful way to train, we seek to address a significant impediment to the widespread use of virtual environments as training systems: acquiring and maintaining the 1) training scenarios, 2) domain knowledge and 3) agent/avatar behaviours. In this paper we focus on the latter two issues.

The goal of this paper was to evaluate if users found MCRDR to be a more comprehensible knowledge representation and acquisition technique than standard production rules and whether providing a scenario context also assisted with knowledge acquisition. In the next section we explain how MCRDR are used in our training simulation application and provide the results of a study showing the efficacy of using MCRDR. We conclude with future work and summary.

2 Acquiring Knowledge and Experience

While the training environment is being developed to assist trainees to acquire the domain knowledge, in this paper we are focused on the comprehensibility of the MCRDR knowledge representation and the usability of the KA process for the human domain expert who will train the system. Two key features of MCRDR which we sought to evaluate is the use of an exception structure and cases to motivate and validate knowledge acquisition [3]. Looking at Fig. 1, rule 1 was added in response to

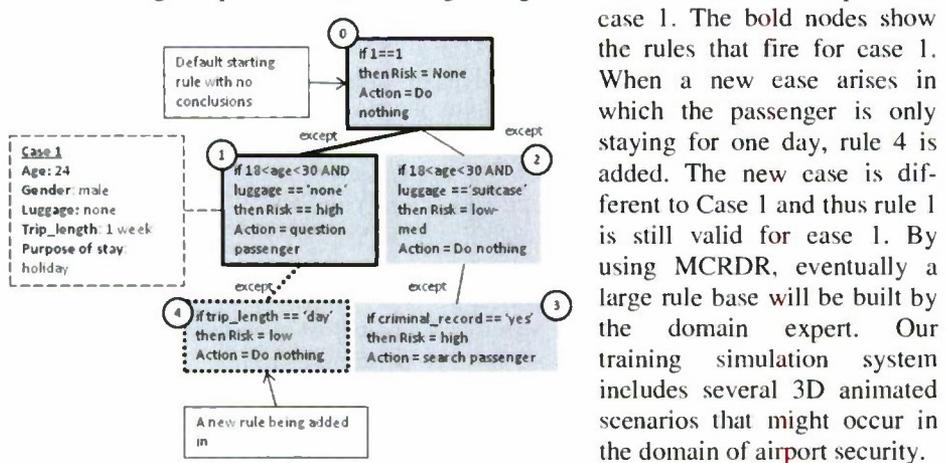


Fig. 1. MCRDR tree with 5 rules and 1 case shown

case 1. The bold nodes show the rules that fire for case 1. When a new case arises in which the passenger is only staying for one day, rule 4 is added. The new case is different to Case 1 and thus rule 1 is still valid for ease 1. By using MCRDR, eventually a large rule base will be built by the domain expert. Our training simulation system includes several 3D animated scenarios that might occur in the domain of airport security.

In our approach, the domain expert can interrupt a running scenario and display its attributes (Fig. 2). At this point the system presents its conclusions for the scenario

based on the RDR KB, initially contains only the default rule not associated with any scenario. The expert may then either agree with the conclusions or disagree by adding a new rule into the RDR KB which will then be associated with that scenario or ease. If they disagree, they will be re-shown the attributes of the current scenario, as well as the attributes of the scenario associated with the rule giving the incorrect conclusion.

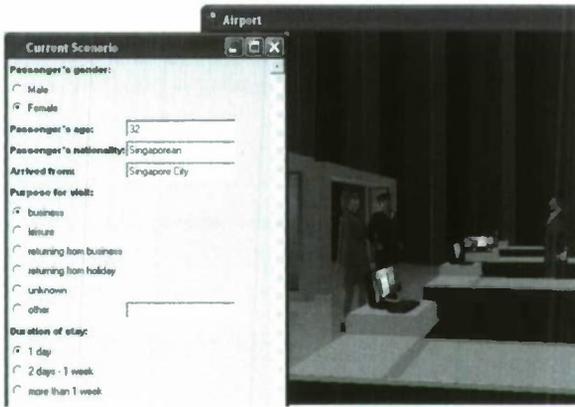


Fig. 2. BOSS screen showing a case being popped up

3 Comprehensibility, Usability and Usefulness Study

We used a 'Repeated Measures' design with two within subjects factors (Scenario, media/format) and one between subjects factor (stimuli order). In the present study this means that all participants received Scenario 1: no luggage, Scenario 2: eriminal conviction and No Scenario in the same experimental session. Furthermore, there were two possible media/formats for presenting the scenarios: the virtual training environment which involved using RDR or in textual format leading to four combinations S1RDR, S1TXT, S2RDR, S2TXT. Each participant encountered both scenarios but received either S1RDR and S2TXT OR S2RDR and S1TXT. In the virtual training environment (VTE) exactly the same text was heard and read as in the text-only treatment. It was our goal to test whether experieneing the seenario in a VTE and using the RDR knowledge acquisition method in that environment was easier, more natural and produced better rules/knowledge.

Each participant was also given the task of writing some production rules without the use of any scenario to provide context. We called this treatment COLD. As we were dealing with novices rather than experts, domain knowledge was provided to participants for each task. Participants thus acted as their own control group. Because

of the increased statistical power of the 'Within Subjects' experiment design, fewer participants were required to draw valid conclusions.

To avoid order effects we altered the order of receiving stimuli. To alloecate treatments to experimental units we used a Latin squares design which controls the variation and

S1RDR	Cold	S2Tx1	2 scenarios (S1, S2)
S2Tx1	S1RDR	Cold	
S2Tx1	S1RDR	Cold	3 formats
S2RDR	Cold	S1Tx1	(RDR, Txt, Cold)
S1Tx1	S2RDR	Cold	
Cold	S1Tx1	S2RDR	

Fig. 3. Experimental Design using Latin square

estimates the main effects of all factors to produce the orders as shown in Fig. 3. A Latin square is an $n \times n$ table containing n different symbols in such a way that each symbol occurs exactly once in each row and exactly once in each column and is used in experimental designs in which one wishes to compare treatments and to control for two other known sources of variation. Three people were assigned to each combination (i.e. 18 participants). Following each treatment participants were asked the questions relevant to that treatment (see Fig. 4).

Scenario 1 - passengers with no luggage
 You are an expert in airport security. Watch/read the first scenario. Then correct the system's conclusions afterwards using the domain knowledge given below.
Knowledge for passengers with no luggage
 1. A passenger with no luggage is immediately suspicious. Customs officers are advised to search the passenger's clothes and body and consider the passenger to present a moderate risk.
 2. If the passenger is only staying for one day, they present a low risk and customs officers should let the passenger through.
Ripple Down Rules Questions
 1. It was easy to understand what the scenario's attributes were.
 2. I found it easy to understand what the system's conclusion was and how to disagree with it.
 3. I found it easy to understand how the system worked out its conclusion.
 4. I found it easy to select extra categories to change the conclusion for the scenario.
 5. Once I had chosen the extra categories, I found it easy to specify what the new conclusion should be.
 6. The user manual provided was important in helping me to understand how to use the system.
Text Questions 1. I understood the scenario attributes 3. It was easy to write rules for the second scenario
 2. It was easy to write rules for the first scenario
Cold Questions 1. It was easy to write the first rule
 2. It was easy to write more rules 3. It was easy to write the example
Comparison questions 1. Which task did you find the easiest?
 2. Which task did you find the hardest? 3. Which task did you find the most enjoyable?

Fig. 4. Sample information and questions in our study

Participants were recruited across campus. The study took one hour. Participants comprised 9 males, 9 females, aged 18-51, average age 22, 11 had a first language other than English (9 Chinese, 1 Indonesian, 1 Korean), 7 were born in Australia, 5 had lived less than 1 year in an English speaking country, half had played computer games for 5-12 years. Descriptive statistics for the RDR questions in Fig. 4 are provided in Table 1. We found no significant difference between S1 and S2 for both the RDR and text treatments. This means that we were able to combine the results of both scenarios to double the number of responses. We see in Fig. 5 that RDR was found to

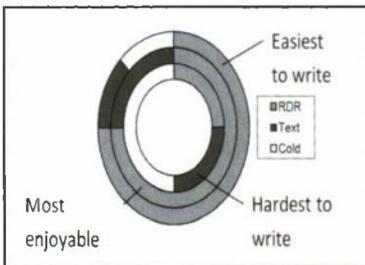


Fig. 5. Results of comparative questions

be the easiest and most enjoyable. Using Median and Mode as measures of central tendency we see that participants understood how to interpret the conditions, rules and conclusions. They also found entering new knowledge easy.

We note the subjectivity of these questions and conducted some analysis on the correctness of the rules. Each rule was scored using the following criteria. For RDR each rule was given a score out of 2 based on; 1) whether they were able to write a rule; 2) whether the rule was

detailed enough based on the knowledge given; 3) whether they showed the features of the RDR in the rule (second rule); and whether the rule was correct based on the knowledge given. Text rules were given a score out of 2 where 1 mark concerned whether the risk was correctly specified and 1 mark considered if the agent action was correctly specified. Cold rules were given a score out of 2 based on whether the rule fits with the knowledge provided; if it fits with other rules (using RDR type logic); number of extra (irrelevant) rules. For each participant a score based on the rank order of treatments was given, with 3 the highest rank. The results are given in Table 2. From the scoring process, and supported in the results, the cold treatment rules tended to lack structure, consistency and relevant content. Providing the context of a scenario in the text treatment was obviously helpful and produced the best rules in this study. The difference in scores for text and cold rules was statistically significant ($p=0.037$). While text ranked highest, we expect that the benefits of RDR for consistency and relevance to the case would be better demonstrated in the longer term (even after a day rather than just 20 minutes of usage) and after some training.

Table 1. Descriptive stats for Likert responses to RDR Qs1-6

Key: 5=Strongly Agree, 4=Agree, 3=Neutral, 2=Disagree, 1=Strongly Disagree.

Q	Ave	StdErr	Med	Mode	Stdev	S/Var	Kurt	Skew	Rnge	Min	Max
1RDR	4.222	0.207	4	5	0.878	0.771	0.868	-1.069	3	2	5
2RDR	4.056	0.189	4	4	0.802	0.644	1.305	-0.875	3	2	5
3RDR	4.111	0.179	4	4	0.758	0.575	1.118	-0.195	2	3	5
4RDR	3.941	0.250	4	5	1.029	1.059	0.546	-0.651	3	2	5
5RDR	3.889	0.227	4	4	0.963	0.928	0.211	-0.645	3	2	5
6RDR	4.167	0.167	4	4	0.707	0.500	0.776	-0.250	2	3	5

Table 2. Comparison of correctness (top score/ave possible is 3)

Anova: Single Factor						
Groups	Count	Sum	Ave	Variance		
RDR	18	37	2.0556	0.6438		
Text	18	44	2.4444	0.6144		
Cold	18	34	1.8889	0.5752		
Var Source	SS	df	MS	F	P-val	F crit
Btween Grps	2.926	2	1.4630	2.3939	0.102	3.1788
Within Grps	31.167	51	0.6111			
Total	34.093	53				

We found a highly statically significant difference for RDR ($p=0.005$) and cold ($p=0.0049$) treatments in the responses according to the order in which the treatment occurred. In general

we can say the first treatment will do worse than the same treatment when done second or third. This finding makes sense given that experience with any task is likely to improve performance. Similarly, the time taken to perform each task was significantly affected by the order in which the task occurred, regardless of the treatment. See in Table 3 that RDR took much longer than the other tasks which were due to the need to read and refer to the user manual. Note that spending more time engaged in a training task is in general beneficial for learning. In performing a correlation between the questions across tasks we find a positive correlation between the question about the

hardest task with questions about understanding the attributes in the Text task (0.856) and in the RDR task (0.701).

Table 3. Comparison of time to conduct each task

Anova: Single Factor							
Groups	Count	Sum	Average	Variance			
RDR time	18	7.10611	0.394784	0.029359			
text time	18	2.45088	0.13616	0.006144			
Cold time	18	2.46188	0.136771	0.004458			
Var Source	SS	Df	MS	F	P-value	F crit	
Btween grps	0.80075		2	0.400372	30.05634	2.38E-09	3.178
Within grps	0.67936		51	0.013321			
Total	1.48010		53				

We also find a medium to strong positive correlation (0.775, 0.778, 0.712) to the question on which task was most enjoyable and Q2, 4 and 5 for

the RDR task. On a F-Test Two-Sample for Variances with CF 95%, there was a statistically significant difference in the responses of participants who played computer games for more than 5 years with those with less gaming experience giving the higher scores.

When determining if there was a significant difference in perceived difficulty of adding the first rule and subsequent consistent rules (a claimed strength of RDR) using one-tail t-Test: Paired Two Sample for Means we found a significant difference ($p=0.036$) showing that participants found adding additional rules more difficult than the first rule for the cold but not for the text treatment.

4 Further Considerations

Gaines [1] proposed the use of an exception directed acyclic graph to measure the comprehensibility of production rules, decision trees and rules with exceptions. The approach computes complexity based on the number of the nodes (N), final/end nodes (F), arcs/edges (A), Excess ($E=A+V-N$), clauses (C), where complexity $X = (N+2E+2C)/5$. Sugiura, Riesenhuber and Koseki [7] also offer the measures of table size, similarity of concept function, continuity in attributes with ordinal values and conformity between concept functions and real cases to determine comprehensibility of tabular knowledge bases. To potentially support better comparison with the RDR rules, we could apply some parsing techniques from language processing (such as link grammar) to generate more structured output from the text and cold rules. In contrast, RDR are structured, linked to cases and linked to scenarios making evolution and growth of the KB possible and supporting Sugiura et al’s (1993) criteria of conformity between concept functions and realistic cases.

We considered asking our participants to represent their knowledge using a logic-based formalism, decision trees or decision tables, however, we did not believe that an untrained person would be able to write valid logic statements, decision tables or trees. We note that RDR can be transformed into a propositional or FOL [4]

representation and also into unambiguous decision tables [1]. We felt that asking participants to write statements in the IF-THEN format (ie production rules) was something we could expect an average person to be able to achieve. Furthermore, in our study we tested the role that context in the form of cases plays in assisting the knowledge acquisition task and thus we provided cases for the text-based scenarios (which were the same as the cases used in the RDR condition) and compared the difficulty of writing rules when this context is not known or specified.

References

1. Colomb, R.M.: Decision Tables, Decision Trees and Cases: Propositional KBS Tech. Rep. 266. Comp. Sci. Dept. UQ, Australia (1993)
2. Gaines, B.R.: Transforming rules and trees into Comprehensible Knowledge Structures. In: Fayyad, U.M., Piatetsky-Shapiro, G., Smyth, P., Uthurusamy, R. (eds.), pp. 205–226. MIT Press, Menlo Park (1996)
3. Kang, B., Compton, P., Preston, P.: Multiple Classification Ripple Down Rules: Evaluation and Possibilities. In: Proc. KAW 1995, February 26-March 3, vol. 1, pp. 17.1–17.20 (1995)
4. Kwok, R.B.: Translations of Ripple Down Rules into Logic Formalisms. In: Dieng, R., Corby, O. (eds.) EKAW 2000. LNCS (LNAI), vol. 1937, pp. 366–379. Springer, Heidelberg (2000)
5. Quinlan, J.R.: Discovering rules by induction from large collections of examples. In: Mitchie, D.E. (ed.) Expert systems in the micro-electronic age. Edinburgh Uni. Press, Edinburgh (1979)
6. Quinlan, J.R.: Fwd. Knowledge Discovery in Databases, pp. ix-xii. MIT Press, Cambridge (1991)
7. Sugiura, A., Riesenhuber, M., Koseki, Y.: Comprehensibility Improvement of Tabular Knowledge Bases. In: AAAI 1993, pp. 716–721 (1993)

Exploiting Comparable Corpora for Cross-Language Information Retrieval

Fatiha Sadat

University of Quebec in Montreal, Computer Science department,
201 President Kennedy avenue, Montreal, QC, Canada
sadat.fatiha@uqam.ca

Abstract. Large-scale comparable corpora became more abundant and accessible than parallel corpora, with the explosive growth of the World Wide Web. Therefore, strategies on bilingual terminology extraction from comparable texts must be given more attention in order to enrich existing bilingual lexicons and thesauri and to enhance Cross-Language Information Retrieval. In the present paper, we focus on the enhancement of Cross-Language Information Retrieval using a two-stage corpus-based translation model that includes bi-directional extraction of bilingual terminology from comparable corpora and selection of best translation alternatives on the basis of their morphological knowledge. The impact of comparable corpora on the performance of the Cross-Language Information Retrieval process is evaluated in this study and the results indicate that the effect is clearly positive, especially when using the linear combination with bilingual dictionaries and Japanese-English pair of languages.

Keywords: Cross-language information retrieval, comparable corpora, similarity, co-occurrence tendency.

1 Introduction

This paper intends to bring solutions to the problem of lexical coverage of existing bilingual dictionaries but also to the improvement of the performance of CLIR. The main contributions concern the enhancement of CLIR by an automatic acquisition of bilingual terminology from comparable corpora that will help cope with the limitation of CLIR, especially in the query disambiguation process as well as during the query expansion with related terms. Furthermore, this study could be valuable for the extraction of unknown words and their translation and thus the enrichment and enhancement of bilingual dictionaries. Therefore, we present in this paper an approach of learning bilingual terminology from textual resources other than bilingual dictionaries, such as comparable corpora and evaluations on CLIR. First, we propose a two-stage corpus-based translation model for the acquisition of bilingual terminology from comparable corpora. The first stage concerns the extraction of bilingual translations from the source language to the target language, also from the target language to the source language. The two results are combined for the purpose of disambiguation. In the second stage, the extracted translation alternatives are filtered on the basis of their

morphological knowledge. A linguistics-based pruning technique is applied in order to compare source words and their target language translation equivalents on the basis of their part of speech tags. Furthermore, we present a combined translation model involving the comparable corpora and readily available bilingual dictionaries. In our evaluations, we used a large-scale test collection on Japanese-English and different weighting schemes of SMART retrieval system and confirmed the effectiveness of the proposed translation model in CLIR.

The remainder of the present paper is organized as follows: Section 2 presents an overview of the proposed model. Section 3 presents the two-stage corpus-based translation model. Section 4 introduces a combination of different translation models. Experiments and evaluations in CLIR are related in Section 5. Section 6 concludes the present paper.

2 An Overview of the Proposed Model

Throughout this paper we will seek to exploit and explore benefits from collections of news articles for the acquisition of bilingual terminology, in order to enrich existing multilingual lexical resources and help cross the language barrier for information retrieval. We rely on such comparable corpora for the extraction of bilingual terminology, in the form of translations and/or expansion terms, i.e. words that will help the query expansion in CLIR. The task of bilingual terminology extraction is accomplished by a two-stage corpus-based translation model, which is described in detail in Section 3. A linear combination involving the comparable corpora and bilingual dictionaries is completed in order to select best translation candidates of the source terms of a given query. Finally, documents are retrieved in the target language.

3 Two-Stage Corpus-Based Translation Model

A two-stage corpus-based translation model (Sadat et al., 2003a; Sadat et al., 2003b; Sadat et al., 2003e), which is based on the symmetrical criterion in addition to the assumption of similar collocation, aims to find translations of the source word in the target language corpus but also translations of the target words in the source language corpus. Linguistic resources were used in the two-stage corpus-based translation model, as follows: (i) a collection of news articles from *Mainichi Newspapers* (1998-1999) for Japanese and *Mainichi Daily News* (1998-1999) for English were considered as comparable corpora, because of their common feature on the time period. Documents of *NTCIR-2* test collection were also considered as comparable corpora in order to cope with special features of the test collection during evaluations; (ii) morphological analyzers, *ChaSen* version 2.2.9 (Matsumoto et al., 1997) for texts in Japanese and *OAK* (Sekine, 2001) for English texts were used in linguistic processing; (iii) *EDR* (1996) and *EDICT*¹ bilingual Japanese-English and English-Japanese dictionaries were considered in the translation of context vectors of source and target languages. Japanese words written in Katakana representing foreign words and proper names, that were not found in the bilingual dictionaries were manually translated. A

¹ <http://www.csse.monash.edu.au/~jwb/wwwjdic.html>

transliteration process could be used in order to convert those words to their English equivalence.

3.1 First Stage in the Proposed Translation Model

The two-stage corpus-based translation model for the acquisition of bilingual terminology is described as follows:

1. A simple bilingual terminology acquisition from source language to target language to yield a first simple translation model represented by similarity vectors $SIM_{S \rightarrow T}$.
2. A simple bilingual terminology acquisition from target language to source language to yield a second simple translation model represented by similarity vectors $SIM_{T \rightarrow S}$.
3. Merge the first and second models to yield a two-stage translation model, based on bi-directional comparable corpora and represented by similarity vectors $SIM_{S \leftrightarrow T}$.

The simple approach for bilingual terminology acquisition from comparable corpora is based on the assumption of similar collocation, i.e., If two words are mutual translations, then their most frequent collocates are likely to be mutual translations as well. We follow strategies of previous researches (Dejean et al., 2002; Fung, 2000; Rapp, 1999; Sadat et al., 2003a; Sadat et al., 2003b, Sadat et al., 2003c).

In further sections, we name the simple approach for bilingual terminology acquisition from comparable corpora as *simple corpus-based translation* and the translation model representing the first stage of the two-stage corpus-based translation as *bi-directional corpus-based translation*.

3.2 Second Stage in the Proposed Translation Model

Combining linguistic and statistical methods is becoming increasingly common in computational linguistics, especially as more corpora become available (Klavens & Tzoukermann, 1996; Sadat et al., 2003c). We propose to integrate linguistic concepts into the corpus-based translation model. Morphological knowledge such as Part-of-Speech (POS) tags, context of terms, etc., could be valuable to filter and prune the extracted translation candidates. The objective of the linguistics-based pruning technique is the detection of terms and their translations that are morphologically close enough, i.e., close or similar POS tags. This proposed approach will select a fixed number of equivalents from the set of extracted target translation alternatives that match the Part-of-Speech of the source term. Japanese foreign words were not pruned with the proposed linguistics-based technique but could be treated via *transliteration*, i.e., conversion of Japanese katakana to their English equivalence or to the alphabetical description of their pronunciation (Knight & Graehl, 1998). Finally, the generated translation alternatives are sorted in decreasing order by similarity values. Rank counts are assigned in increasing order, starting at 1 for the first sorted list item. A fixed number of top-ranked translation alternatives are selected and misleading candidates are discarded.

4 Combining Different Translation Models

Combining different translation models has showed success in previous research (Dejean et al., 2002). We propose a combined translation model involving comparable corpora and readily available bilingual dictionaries. The proposed dictionary-based translation model is derived directly from readily available bilingual dictionaries, by considering all translation candidates of each source entry as equiprobable, to yield a probabilistic translation model $P_2(t|s)$. The linear combination will involve the two probabilistic translation models $P_1(t|s)$ and $P_2(t|s)$ derived from the comparable corpora (either the simple or the two-stage model) and readily available bilingual dictionaries.

5 Evaluation and Experiments

We considered the set of news articles as well as the abstracts of NTCIR-2 test collection as comparable corpora for Japanese-English language pairs. Content words (nouns, verbs, adjectives, adverbs and Foreign words) were extracted from English and Japanese corpora. Context vectors were constructed for 13,552,481 Japanese terms and 1,517,281 English terms. Similarity vectors were constructed for 96,895,255 (Japanese, English) pairs of terms and 92,765,129 (English, Japanese) pairs of terms. Bi-directional similarity vectors (after merging and disambiguation) resulted in 58,254,841 (Japanese, English) pairs of terms. *SMART* information retrieval system (Salton, 1971), which is based on vector model, was used to retrieve English documents. We used the monolingual English runs, i.e., English queries to retrieve English documents and the bilingual Japanese-English runs, i.e., Japanese queries to retrieve English documents. Bilingual translations were extracted from the collection of news articles using the simple translation model and the two-stage translation model. A fixed number p (set to five) of top-ranked translation alternatives was retained for evaluations in CLIR. Results and performances on the monolingual run as well as on the bilingual runs using the two-stage corpus-based translation model and the linear combination to bilingual dictionaries are illustrated in Table 1. Evaluations are based on the average precision, differences in term of average precision of the monolingual counterpart and the improvement over the monolingual counterpart.

Retrieval methods are represented by the monolingual retrieval *Mono*, dictionary-based translation *DT*, the simple corpus-based translation model *SCT*, the bidirectional corpus-based translation model *BCT*, the two-stage corpus-based translation model *TCT*. Linear combinations were represented by *SCT+DT*, *BCT+DT* and *TCT+DT*.

Table 1. Evaluations of the proposed and combined translation models

Average Precision, and % Monolingual ($P=5$)							
<i>Mono</i>	<i>DT</i>	<i>SCT</i>	<i>BCT</i>	<i>TCT</i>	<i>SCT+DT</i>	<i>BCT+DT</i>	<i>TCT+DT</i>
0.3368 (100%)	0.2279 (67.66%)	0.1417 (42.07%)	0.1801 (53.47%)	0.2008 (59.62%)	0.2366 (70.25%)	0.2721 (80.79%)	0.2987 (88.69%)

As illustrated in Table1, combining different translation models yields a significantly better result than using each model by itself. Translation models based on comparable corpora and bilingual dictionaries have completed each other and their linear combination has provided a valuable resource for query translation/expansion in CLIR and has allowed an improvement in the effectiveness of information retrieval.

6 Conclusion

In the present paper, we investigated the approach of extracting bilingual terminology from comparable corpora in order to enhance CLIR, especially in the disambiguation and query expansion processes, and possibly enrich existing bilingual lexicons. We proposed a two-stage corpus-based translation model consisting of bi-directional extraction of bilingual terminology and linguistic-based pruning. Among the drawbacks of the proposed translation process is the introduction of many noisy terms or wrongly translated terms; however, most of those terms could be considered as efficient for the query expansion in CLIR but not for the translation.

Combination of two-stage corpus-based translation model and bilingual dictionaries yields to better translations and an effectiveness of information retrieval could be achieved across Japanese and English languages.

References

1. Dejean, H., Gaussier, E., Sadat, F.: An Approach based on Multilingual Thesauri and Model Combination for Bilingual Lexicon Extraction. In: Proceedings of COLING 2002, Taiwan, pp. 218–224 (2002)
2. EDR: Japan Electronic Dictionary Research Institute, Ltd. EDR electronic dictionary version 1.5 technical guide. Technical report TR2-007. Japan Electronic Dictionary research Institute, Ltd. (1996)
3. Fung, P.: A Statistical View of Bilingual Lexicon Extraction: From Parallel Corpora to Non-Parallel Corpora. In: Véronis, J. (ed.) Parallel Text Processing (2000)
4. Klavens, J., Tzoukermann, E.: Combining Corpus and Machine-Readable Dictionary Data for Building Bilingual Lexicons. *Machine Translation* 10(3-4), 1–34 (1996)
5. Knight, K., Graehl, J.: Machine Transliteration. *Computational Linguistics* 24(4) (1998)
6. Matsumoto, Y., Kitauchi, A., Yamashita, T., Imaichi, O., Imamura, T.: Japanese morphological analysis system ChaSen manual. Technical report NAIST-IS-TR97007, NAIST (1997)
7. Rapp, R.: Automatic Identification of Word Translations from Unrelated English and German Corpora. In: Proceedings of European Chapter of the Association for Computational Linguistics, EACL (1999)
8. Sadat, F., Yoshikawa, M., Uemura, S.: Enhancing Cross-language Information Retrieval by an Automatic Acquisition of Bilingual Terminology from Comparable Corpora. In: Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2003, Toronto, Canada (2003)

9. Sadat, F., Yoshikawa, M., Uemura, S.: Learning Bilingual Translations from Comparable Corpora to Cross-Language Information Retrieval: Hybrid Statistics-based and Linguistics-based Approach. In: Proceedings of the Sixth International Workshop on Information Retrieval with Asian Languages, IRAL 2003, Sapporo, Japan (2003)
10. Sadat, F., Yoshikawa, M., Uemura, S.: Bilingual Terminology Acquisition from Comparable Corpora and Phrasal Translation to Cross-Language Information Retrieval. In: Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics, ACL 2003, Sapporo, Japan (2003)
11. Sadat, F.: Knowledge Acquisition from Collections of News Articles to Cross-language Information Retrieval. In: Proceedings of RIAO 2004 conference (Recherche d'Information Assisté par Ordinateur), Avignon, France, April 26-28, pp. 504-513 (2004)
12. Salton, G.: The SMART Retrieval System, Experiments in Automatic Documents Processing. Prentice-Hall, Inc., Englewood Cliffs (1971)
13. Salton, G., McGill, J.: Introduction to Modern Information Retrieval. Mc Graw-Hill, New York (1983)
14. Sekine, S.: OAK System- Manual. New York University (2001)

Local PCA Regression for Missing Data Estimation in Telecommunication Dataset

T. Sato, B.Q. Huang, Y. Huang, and M.-T. Kechadi

School of Computer Science and Informatics, University College Dublin, Belfield,
Dublin 4, Ireland

bingquan.huang@ucd.ie, takeshi.sato@ucdconnect.ie

Abstract. The customer churn problem affects hugely the telecommunication services in particular, and businesses in general. Note that in majority of cases the number of potential customer churn is much smaller than the non-churners. Therefore, the imbalance distribution of samples between churners and non-churners is a concern when building a churn prediction model. This paper presents a Local PCA approach to solve imbalance classification problem by generating new churn samples. The experiments were carried out on a large real-world Telecommunication dataset and assessed on a churn prediction task. The experiments showed that the Local PCA along with Smote outperformed Linear regression and Standard PCA data generation techniques.

Keywords: PCA, Imbalanced Classification, Churn Prediction.

1 Introduction

Customer Churn has become a serious problem for companies mainly in telecommunication industry. This is as a result of recent changes in the telecommunication industry, such as, new services and the liberalisation of the market. In recent years, Data Mining techniques have emerged as one of the method to tackle the Customer Churn problem[1,8].

The study of customer churn can be seen as a classification problem (Churn and Non-Churn classes). The main goal is to build a robust classifier to predict potential churn customers. However, imbalanced distribution of class samples is an issue in data mining as it leads to poor classification results[4]. In this paper, we focus on overcoming this problem by increasing the size of churn samples by an over-sampling approach. The aim is to correctly set the distribution samples to build an optimal classifier by adding minority class samples. There have been various sampling approaches proposed to counter non-heuristic sampling problems.

Synthetic Minority Over-sampling Technique (Smote)[3] generates artificial data along the line between minor class samples and K minority class nearest

neighbours. This causes the decision boundaries for the minor class space to spread further into majority class space. An extended approach of Smote is Sinote + Edited Nearest Neighbour (ENN)[9] approach, which removes more unnecessary samples and provides a more in depth data cleaning.

The main idea of our approach is to form a new minority class space by generating minority class data using the K-means algorithm with PCA[7]. PCA reveals the internal structure of a dataset by extracting uncorrelated variables known as Principal Components (PC). In this paper, we adopt Local PCA data regression to generate new dataset and add raw data to change the distribution of class samples.

This paper is organised as follows: the next section outlines the proposed approach on churn prediction task. Section 3 explains experiments and the evaluation criteria. We conclude and highlight some key remarks in Section 4.

2 Approaches

Our proposed approach combines PCA technique, the Genetic Algorithm (GA) and K-means algorithm to generate a new data for minority class. First and foremost, minority class dataset d_{churn} is formed from the original raw dataset d_{raw} . The GA K-means clustering is applied on d_{churn} to form K clusters. The next step is to apply PCA regression on each cluster set to transform them back to original feature space in terms of selected principal component. We believe that applying regression locally would avoid the inclusion of redundant information in principal component because of lower variance within the clusters. These transformed data is then added to d_{raw} to improve the distribution of minority class samples. Finally, d_{raw} is used to build a churn prediction model for a classification purpose. Figure 1 shows the main steps of the approach.

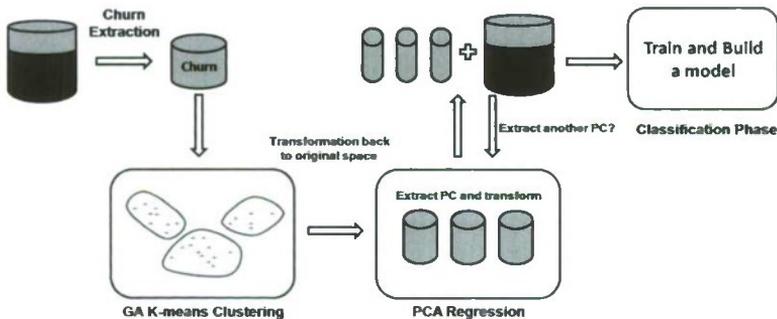


Fig. 1. The description of the proposed approach

2.1 GA K-Means Clustering Algorithm

The standard K-means algorithm is sensitive to the initial centroid and poor initial cluster centres would lead to poor cluster formation. We employ Genetic Algorithm (GA)[5] to avoid sensitivity problem in centroid selection.

In GA K-means algorithm, a gene represents a cluster centre and a chromosome of K genes represents a set of K cluster centres. The GA K-means algorithm steps are: 1) **Initialization**: Randomly select K points as cluster centres (chromosomes) from original data set and apply k-means, 2) **Selection**: The chromosomes are selected according to specific selection method, 3) **Crossover**: Selected chromosomes are randomly paired with other parents for reproduction, 4) **Mutation**: Apply mutation operation to ensure diversity in the population, 5) **Elitism**: Store the chromosome that has the best fitness value in each generation and 6) **Iteration**: Go to step 2, until the variation of fitness value within the best chromosomes is less than a specific threshold.

2.2 Linear PCA and Data Generation

We apply the PCA regression technique on each cluster to generate a new dataset in original feature space in terms of selected principal components (PC). PCA has a property of searching for PC that accounts for large part of total variance in the data and projecting data linearly onto new orthogonal bases using PC.

Consider a dataset $X = \{x_i, i = 1, 2, \dots, N, x_i \in \mathbb{R}^N\}$ with attribute size of d and N samples. The data is standardised so that the standard deviation and the mean of each column are 1 and 0, respectively. PC can be extracted by solving the following Eigenvalue Decomposition Problem[7].

$$\lambda\alpha = \mathbf{C}\alpha, \text{ subject to } \|\alpha\|_2 = \frac{1}{\lambda} \quad (1)$$

where α is the eigenvectors and \mathbf{C} is the covariance matrix. After solving the equation (1), sort the eigenvalues in descending order as larger eigenvalue gives significant PC. Assume that matrix α contains *only* a selected number of eigenvectors (PC). The transformed data is computed by

$$X_{tr} = \alpha^T X^T \quad (2)$$

From equation (2), matrix X^T can be obtained by $X^T = \alpha^{T^{-1}} X_{tr}$. Finally, the matrix X^T is transposed again to get the matrix X^{new} . Since we standardised the data in the first step, the original standard deviation and the mean of each column must be included in each X_{ij}^{new} . The newly generated data X^{new} is then added to d_{raw} to adjust the distribution of the samples. We continue this process until all clusters are transformed.

We run two data generation approaches on PCA regression. The first approach utilises all clusters on PCA regression (Local1). The second approach only uses the centroid of each cluster to form a dataset of centre points (Local2) and use this data to extract principal components.

3 Experiments

3.1 Dataset and Evaluation Criteria

We selected randomly 139,000 customers from a real world database provided by Eircom. The distribution of churn and non-churn is imbalanced as the training and testing data contain 6000 (resp. 2000) churners and 94000 (resp. 37000) non-churners, respectively. These datasets are described by 122 features which are explained in [6].

We implement the Decision Tree C4.5 (DT), the SVM, Logistic Regression (LR) and the Naive Bayes (NB) to build prediction models. We performed these models following the evaluation criteria: 1) The true churn (**TP**) is the ratio of churn that was classified correctly and 2) the false churn (**FP**) is the ratio of non-churn that was incorrectly classified as churn. A good solution is considered as dominant when TP is high and FP is low. We use the Receiver Operating Curve technique (ROC) to evaluate the various learning algorithms. It was shown how that the TP varies with FP. In addition, the Area under ROC curve (AUC)[2] provides single number summary for the performance of learning algorithms. We calculate the AUC threshold on FP as 0.5 as telecom companies are generally not interested in FP above 50%.

3.2 Experimental Setup

The main objective of the experiments is to observe if additional churner samples generated by PCA regression would improve churn prediction results. We first examines the optimal cluster size of the GA K-means algorithm for Local PCA regression by setting K to be in $[4 - 72]$. The second experiment compares the prediction results of each classifier by PCA regression from experiment 1 to Linear Regression(LiR), Standard PCA based data generation and Smote. The final experiment examines the main objective. The number of churners is increased from the original size of 6000 up to 30000 by setting the PC threshold to 0.9, 0.8, ..., 0.6. A new dataset is generated based on Local1 & Local2 generation method for all experiments.

3.3 Results and Discussion

The range of cluster size K , [36:72], produced better AUC results than smaller K . In addition, GA K-means performed generally better than standard K-means.

The FP and TP rates of 3 data regression methods and Smote were compared in Figure 2. For local PCA, we selected 2 best cluster sizes from range [36:72] for each classifier. The standard PCA operates similarly to local PCA but the clustering technique is not applied on churn data. For all classifiers, both types of PCA regression performed as good as Smote and better than other methods except C4.5, as it is hard to conclude which method is better.

The third experiment overall results are illustrated in Figure 3. The Figure presents the graphs of AUC against the churn size for each classifier using *Local1*

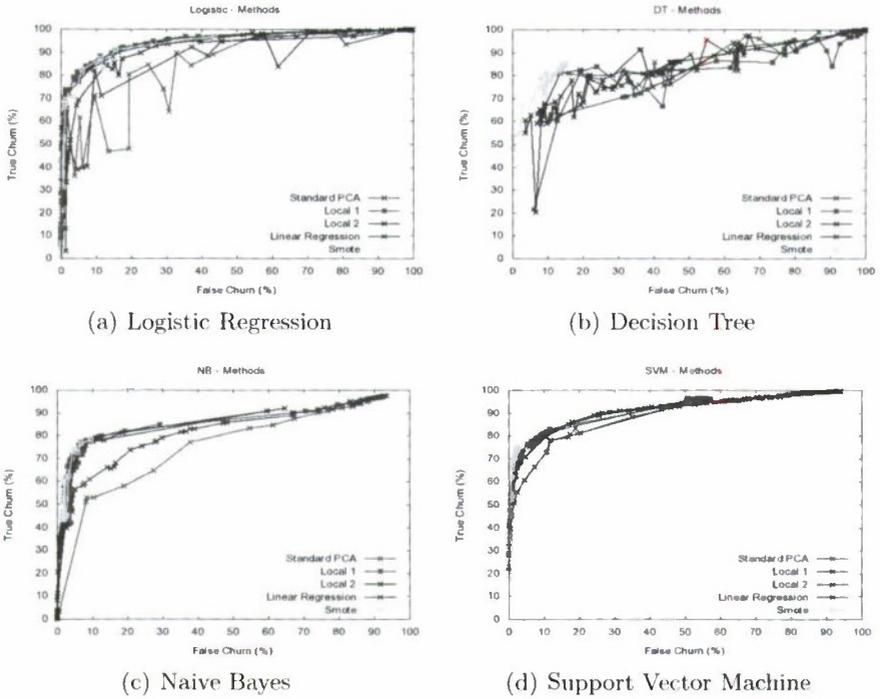


Fig. 2. ROC graph: Comparison of Local PCA, Standard PCA and Linear Regression

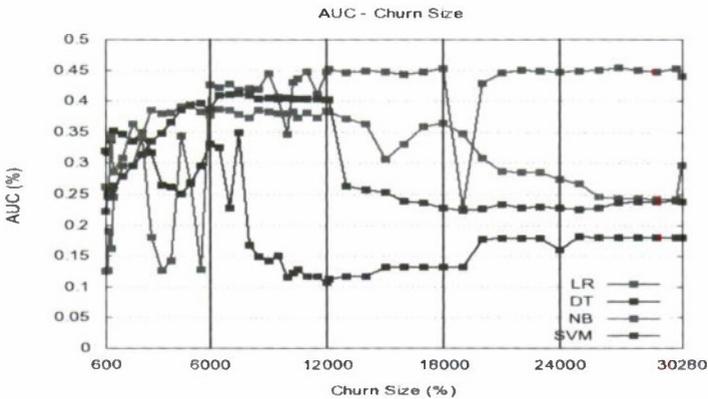


Fig. 3. AUC Plot

data generation as it gave the best prediction results in experiment 2. From the churn size of 6000 onward, additional churn samples generated by PCA were added. The SVM, NB and LR performed well with size 6000 to 12000 but they did not produce acceptable TP or FP rates afterwards as this can be easily seen from Figure 3.

In summary, the experiments showed that 1) The clustering size K did produce different AUC results according to the size, 2) The local PCA data regression performed better than Standard PCA and LiR and finally 3) Adding similar churn samples to original data improved the TP rate for most of the classifiers. However, FP reached over 50% after 12000. One of the reasons for the high FP rate is due to the change in decision boundaries. More non-churn samples inside enlarged churn space lead to high number of incorrectly classified non-churn.

4 Conclusion and Future Works

In this paper, we have designed PCA regression method locally in combination with GA K-means algorithm to generate churn class samples in anticipation to solve Imbalance classification problem.

The approach was tested on a telecommunication data on churn prediction task. The results showed that the Local PCA along with Smote performs better than Standard PCA and LiR in general. Additional samples would improve TP rate for churn size [6000:12000] but the FP rate would increase over 50%. Since we are more interested in identifying potential churner as losing a client has significant effect for the telecom company, improvement in TP is a good results. Nevertheless, FP rate must be limited as high FP can be expensive for future marketing campaign. We are interested in understanding as to why additional churn samples would give high FP. There is a possibility that the churn data generated by various PC thresholds can lead to poor classification in FP.

References

1. Au, W., Chan, C.C., Yao, X.: A novel evolutionary data mining algorithm with applications to churn prediction. *IEEE Transactions on Evolutionary Computation* 7, 532–545 (2003)
2. Bradley, A.P.: The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern Recognition* 30, 1145–1159 (1997)
3. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kergelmeyer, W.P.: Smote: synthetic minority over-sampling technique. *JAIR* 16, 321–357 (2002)
4. Chawla, N.V., Japkowicz, N., Kotcz, A.: Editorial: special issue on learning from imbalanced data sets. *SIGKDD Explor. Newsl.* 6(1), 1–6 (2004)
5. Goldberg, D.E.: *Genetic Algorithms in Search, Optimization and Machine Learning*. Kluwer Academic Publishers, Dordrecht (1989)
6. Huang, B.Q., Kechadi, M.-T., Buckley, B.: Customer churn prediction for broadband internet services. In: Pedersen, T.B., Mohania, M.K., Tjoa, A.M. (eds.) *DaWaK 2009*. LNCS, vol. 5691, pp. 229–243. Springer, Heidelberg (2009)
7. Jolliffe, I.T.: *Principal Components Analysis*. Springer, Heidelberg (1986)
8. Wei, C., Chin, I.: Turning telecommunications call details to churn prediction: a data mining approach. *Expert Systems with Applications* 23, 103–112 (2002)
9. Wilson, D.L.: Asymptotic properties of nearest neighbor rules using edited data. *IEEE Transactions on Systems, Man and Communications* 2(3), 408–421 (1972)

An Influence Diagram Approach for Multiagent Time-Critical Dynamic Decision Modeling

Le Sun¹, Yifeng Zeng², and Yanping Xiang¹

¹ School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, P.R. China

² Department of Computer Science, Aalborg University, DK-9220 Aalborg, Denmark
sunle2009@gmail.com, yfzeng@cs.aau.dk, xiangyanping@gmail.com

Abstract. Recent interests in multiagent dynamic decision modeling in partially observable multiagent environments have led to the development of several representation and inference methods. However, these methods have limited application under time-critical conditions where a trade-off between model quality and computational tractability is essential. We present a formal representation for modeling time-critical multiagent dynamic decision problems through interactive dynamic influence diagrams. The proposed model, called interactive time-critical dynamic influence diagrams, has the ability to represent space-temporal abstraction in multiagent dynamic decision models. More importantly, we take the notion of object-orientation design which facilitates the self-expansion and self-compression in the model implementation.

Keywords: Time-Critical Decision Making, Multiagent Systems, Model Construction.

1 Introduction

There is a growing line of interest for addressing single agent time-critical dynamic decision problems [3,7]. Time-critical decision modeling is more significant for multiagent applications due to the complex decision process and solutions. Our interest in time-critical multiagent systems is motivated by the emergence of several applications including anti-air defense domain [5], Robocup [4] and multi-player online games [8]. Additionally, a suitable set of time-critical decision making techniques would allow multiple agents to coordinate their actions within a time limit so that individual rational actions do not adversely affect the overall system efficiency [1].

The purpose of this paper is to present a form technique, called *Interactive time-critical dynamic influence diagrams* (I-TCDIDs) for modeling multiagent time-critical dynamic decision problems. We rest on the representation of *interactive dynamic influence diagrams* (I-DIDs) [2], and further formalize I-DID by providing time-index for each node in the model which follows the same vein as time-critical dynamic influence diagrams [7]. The modeling of time is often reasonable, but what we would really like is a flexible modeling language to

simplify models of problems with several repetitive structures especially for the case that models need to be expanded over time. We therefore take the notion of object-orientation to design an efficient representation scheme for I-TCDDs. The proposed design reduces the implementation complexity of the problem and makes possible the models self-expansion and self-compression.

2 Background

I-DID provides a relatively efficient method for representing multiagent sequential decision problems [2]. Its static model, called interactive influence diagram (I-ID), extends influence diagrams by introducing a new model node. We show one example of I-ID in Fig. 1(a). The I-ID model is constructed from the viewpoint of agent i that interacts with agent j . The model node, $M_{j,l-1}$, contains possible computable models of other agent like $m_{j,l-1}^1, \dots, m_{j,l-1}^n$ in the low level $l-1$. Solutions of all models are weighted by agent i 's beliefs on j 's models and aggregated into chance node A_j (via the policy link). The issue becomes complicated when I-ID is expanded into I-DID over time. As agent j may act and receive observations, its models need to be updated to reflect the new beliefs. We assume the model node at time t , $M_{j,l-1}^t$, contains two j 's models ($m_{j,l-1}^{t,1}$ and $m_{j,l-1}^{t,2}$), and show the model update in Fig. 1(b). Since agent j may receive any of $|O_j| (=2)$ possible observations the updated set at time $t+1$ will become 4 models ($m_{j,l-1}^{t+1,1}, \dots, m_{j,l-1}^{t+1,4}$). The four models differ in their initial beliefs. The distribution over the updated set of models in the chance node $Mod[M_j^{t+1}]$ depends on the distributions over j 's action and observation that led to these models, and the prior distribution over the models at time step t . More details about I-DID refers to [2] due to the limited space here.

Current design of I-DIDs or most probabilistic graphical models are not essentially rooted in the object-oriented paradigm. We perceive that object-orientation conception would improve the current design and implementation. Here it is necessary to cover some of basic concepts. In the object-oriented paradigm the basic component is an object, an instance of a class. A class is a description of objects with common structures, behaviors and attributes, and has an associated set of nodes, connected by links. In addition to usual nodes

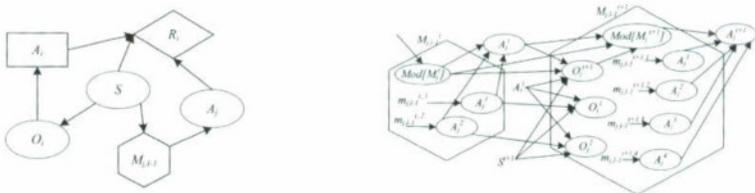


Fig. 1. (a) A generic level $l > 0$ I-ID for agent i with a model node ($M_{j,l-1}$) and the policy link represented by the dashed arrow. (b) Model update from t to $t+1$. $Mod[M_j^t]$ has the number of j 's models as its values. Notice the growth of models in the model node at $t+1$ in bold.

in probabilistic graphical models, a class may also contain special nodes, called instance nodes, representing instances of other classes. A class instance represents a network containing three sets of nodes as defined in HUGIN: input nodes, output nodes and protected nodes. Input nodes and output nodes are the class interfaces and used to link the class instances to other network fragments. They must only be decision or chance nodes. Protected node is the node that only has parents and children inside the class. It can be all kinds of nodes.

3 Interactive Time-Critical Dynamic Influence Diagrams(I-TCDDs)

I-TCDD extends I-DID by including the concepts of temporal arcs and time sequences. Furthermore, three classes(agent class, time-slice class and inference class) are defined in I-TCDDs to efficiently avoid repetition of identical structures. The use of object-oriented conception realize models self-expansion and self-compression for complex problems.

Each node in an I-TCDD represents a set of time-indexed variables. Arcs in an I-TCDD are called temporal arcs and denote both probabilistic and temporal (time-lag) relations among the variables. I-TCDD allows for the coexistence of nodes with different temporal information in the same model.

Often problems in dynamic multiagent domain are of a repetitive nature, such as different agents of the same type and several time-slices. Naturally, these repeated structure should be modeled using agent class and time-slice class. As mentioned in Section 2, I-DIDs introduce a specific model node representing other agents models and the models are expanded over time. This would become inflexible and redundant while many agents are considered. I-TCDDs address this gap by allowing the representation of other agents' models as the values of instances of agent class(agent instances). Time-slice class is a fragment continuously repeated with links between the slices representing different time intervals. As the same as [6], an outer-most class, called inference class in our work, should be defined to provide additional information and perform inference. Performing an instantiation of the inference class gives us the equivalent of a DID, which makes it necessary to use ordinary DID inference engines.

3.1 Agent Class

Agent class models common domain structures, behaviors and attributes in the domain. An instance of agent class includes a set of agent instances and some usual nodes to assist inference and result in an optimal strategy as the output. A specific agent, say j 's instance node is shown in Fig. 2. It interacts with the surroundings by an input node(an oval with heavy grey border) and an output node(an oval with gray filling color). The values of input node S represents a set of current states while output node $A_{j,l-1}$ is a set of optimal actions of agent j . The nodes $m_{j,l-1}^1, \dots, m_{j,l-1}^n$, are agent j 's alternative computational models ascribed by i in level $l - 1$. Each computational model is an instance of agent

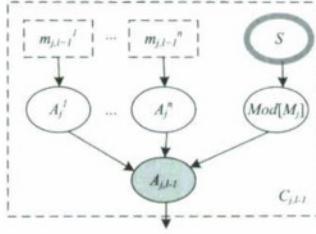


Fig. 2. The detailed instance node of agent j with several computational models ($m_{j,t-1}^1, \dots, m_{j,t-1}^n$) instantiated from agent class

j in the low level. Hence agent class is defined in a recursive way. For several agents in an interacting environment, we could have an agent instance for every agent.

3.2 Time-Slice Class and Inference Class

The basic building block of I-TCDD is a one time-interval network fragment of a specific domain. It is an instance of a class called time-slice class. Fig. 3 shows a time-slice class with four wanted time-intervals. The input nodes are place-holders of variables in the previous time step, while the output nodes represent a set of corresponding variables at the current time step. Solid arcs are instantaneous arcs and dashed arcs are time-lag arcs that model relationships between nodes in continuous time-slices. For instance, the dashed arc between S^t and S^{t-1} represents the physical states in current time-slice influencing that of next time-slice.

The nodes $C_{j,l-1}^{t-1}$ and $C_{j,l-1}^t$ are not the actual interface nodes. The arcs, coming from and going to the agent instance node, are called *influential arcs* only representing the influential relationships between the father and the child. The solid bold arc from $C_{j,l-1}^{t-1}$ to $C_{j,l-1}^t$ is a new arc called *model update arc*

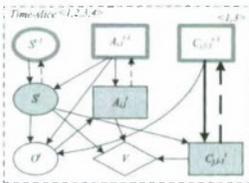


Fig. 3. A generic level l time-slice class for agent i . Notice the model update arc represented by solid bold arc denotes the update of the models of j and of the distribution over the models over time.

Fig. 4. An agent instance node in which two models, m_j^1 and m_j^2 , have different sub-time sequences $\langle 1, 3 \rangle$ and $\langle 1, 2 \rangle$ respectively.

(the time-indexed model update link [2]) reflecting updates of models in agent instance node between two continuous time-slices. The updated model node demands only the place-holders S and instances of agent j 's classes e.g. $m_{j,l-1}^t$ and so on. The influential arc and model update arc may be both replaced by the arcs between the usual nodes.

Then we focus on the conception of *master-time sequence* and *sub-time sequence* which is used to realize time-abstraction. The master-time sequence represents the wanted time-intervals in the modeling process. A sub-time sequence is a subset of master-time sequence and is used to reduce unnecessary information at specific time steps. In Fig. 3, the node $C_{j,l-1}^{t-1}$ is indexed by sub-time sequence $\langle 1, 3 \rangle$ while others are indexed by master-time sequence $\langle 1, 2, 3, 4 \rangle$. For simplicity, we don't show the master-time sequence of the nodes.

Recall that the agent instance node contains all candidate models of other agents. These models may themselves be agent instances leading to recursive modeling. They may be abstracted in a different way. This requires to index each model with a unique time sequence in the agent instance node. Assume that agent j has two candidate models, m_j^1 and m_j^2 . we show one example of an agent instance node with different time-indexed models in Fig. 4. In this case, model m_j^1 is indexed by the sub-time sequence $\langle 1, 3 \rangle$ while m_j^2 is indexed by $\langle 1, 2 \rangle$. We may also index the instance node using a single time sequence if all models share the same sequence. This is exactly the case in Fig. 3 where $C_{j,l-1}^{t-1}$ is time-indexed by $\langle 1, 3 \rangle$ and all models have the same time sequence $\langle 1, 3 \rangle$. In this case, Agent j may not be considered in time sequence $\langle 2, 4 \rangle$ for its negligible influence. Agent j may take actions for fewer time steps and play an intervention only at the indexed times. This means that agent j has been temporally abstracted by omitting its value at some intermediate time indices.

Fig. 5 shows the deployed process of the time-slice class described in Fig. 3. We repeat a normal node (except the instance node) only if its time sequence is equivalent to the master-time sequence; otherwise, it will be casted into a deterministic node (which is deterministically dependent on its parent nodes) for the time step where the index value is omitted from the time sequence. For the agent instance node, we update the model only at the time step if the time is indexed in the time sequence to the model inside the model node. Otherwise, we retain all models from the previous time step and do not perform any model update - we also mark the instance node using the type of deterministic nodes. There is no solutions (actions performed by agents) from the model at a particular time step which is not indexed in the time sequence. For facilitating the CPT setting of action node $A_{j,l-1}$, we assume a uniform distribution of actions from the model, e.g. assigning the probability $\frac{1}{|A_j|}$ to the columns corresponding to the model.

Instances of time-slice class should be encapsulated by an outer-most class, called inference class here, to perform inference. The selected initial information of inference class(Fig. 6) can be input into variables of time-slice class.

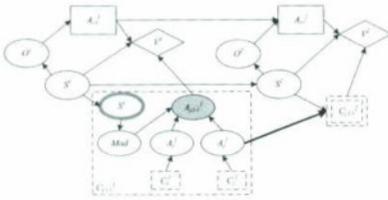


Fig. 5. The deployed form of time-slice class only with two time-slices. The instance node $C_{j,l-1}^2$ is represented by a deterministic node. The instances C_j^1 and C_j^2 are computational models with different beliefs and yet identical time-index.

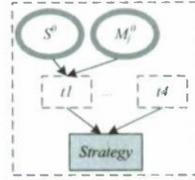


Fig. 6. An inference class with input information of time-slice class. The input node M_j^0 represents the initial belief of agent j 's models. The optimal strategy can be obtained by the *Strategy* node.

4 Conclusion

We propose a formal model of I-TCDDs to represent multiagent time-critical dynamic decision problems. The new technique uses an object-orientation concept to abstract the representation especially on the model expansion over time. It defines an instance of inference and time-slice class based on the concept of agent class. Future work would be interesting to study the impact of initialization on the inference instance in I-TCDDs.

Acknowledgement

Yifeng Zeng acknowledges the partial support from National Natural Science Foundation of China (No. 60974089 and No. 60975052). Both Le Sun and Yan-ping Xiang thank the support from Natural Science Foundation of China (No. 60974089).

References

1. Bond, A., Gasser, L.: Readings in Distributed Artificial Intelligence. Morgan Kaufmann, San Mateo (1988)
2. Doshi, P., Zeng, Y., Chen, Q.: Graphical models for interactive pomdps: Representations and solutions. *Journal of Autonomous Agents and Multiagent Systems* 18, 376–416 (2009)
3. Horvitz, E., Seiver, A.: Time-critical action: Representations and application. In: *Proceedings of the Thirteenth Conference on Uncertainty in Artificial Intelligence*, pp. 250–257 (1997)
4. Kitano, H., Kuniyoshi, Y., Noda, I., Asada, M., Matsubara, H., Osawa, E.: Robocup: A challenge problem for ai. *AI Magazine* 18, 73–85 (1997)
5. Noh, S., Gmytrasiewicz, P.J.: Agent modeling in anti-air defense: A case study. In: *Proceedings of the Sixth International Conference in User Modeling*, pp. 389–400 (1997)

6. Bangs, Ø., Olesen, K.G.: Applying object oriented bayesian networks to large medical decision support systems. In: Proceedings of 8th Scandinavian Conference on Artificial Intelligence, pp. 25–36 (2003)
7. Xiang, Y., Poh, K.: Time-critical dynamic decision making. In: Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence, pp. 688–695 (1999)
8. Yee, N.: The demographics, motivations, and derived experiences of users of massively multi-user online graphical environments. *Presence: Teleoperators and Virtual Environments* 15, 309–329 (2006)

Active Learning for Sequence Labelling with Probability Re-estimation

Dittaya Wanvarie¹, Hiroya Takamura², and Manabu Okumura²

¹ Department of Computational Intelligence and Systems Science
dittaya@lr.pi.titech.ac.jp

² Precision and Intelligence Laboratory
Tokyo Institute of Technology

4259 Nagatsuta-cho, Midori-ku, Yokohama City, Japan
{takamura,oku}@pi.titech.ac.jp

Abstract. In sequence labelling, when the label of a token in the sequence is changed, the output probability of the other tokens in the same sequence would also change. We propose a new active learning framework for sequence labelling which take the change of probability into account. At each iteration of the proposed method, every time the human annotator manually annotates a token, the output probabilities of the other tokens in the sequence are re-estimated. This proposed method is expected to reduce the amount of human annotation required for obtaining a high labelling performance. Through experiments on the NP chunking dataset provided by CoNLL, we empirically show that the proposed method works well.

Keywords: active learning, sequence labelling, semi-supervised learning, partial annotation, re-estimation.

1 Introduction

Many natural language processing tasks such as base NP chunking, named entity recognition, semantic role labelling, can be regarded as sequence labelling tasks. The sequence labelling task is a task to assign an output label to each token in the given input sequence. The accuracy of sequence labelling depends on the feature set design, the labelling algorithm, and also the quality of the training set. In order to obtain a good accuracy, we need a considerably large size of labelled data which can only be obtained by expensive human annotation. In order to reduce the amount of human annotation, active learning has been proposed in [1]. In active learning for sequence labelling, the system automatically selects yet-unlabelled informative training sequences and asks the human annotator to annotate the sequences. Hence the system can often achieve high accuracy with a relatively small amount of human annotation work.

In sequence labelling, each output label in a sequence is predicted with different confidence. If the system is uncertain in predicting the label of a token, we should manually annotate the token. On the other hand, we can let the system automatically annotate the other tokens. This idea was implemented by

Tomanek and Hahn [2]. In their method, if the marginal probability of the predicted label of a token is low, this token is manually annotated, and the labels of the other tokens with high probability remain unchanged. They succeeded in reducing the required amount of human annotation. However, we would like to point out that if a token is manually labelled, the probability of the output itself is changed and also affects the probability of labels of other tokens in its neighbourhood since the labels are usually dependent on each other in sequence labelling. If the changed probability exceeds the confidence threshold, the system can automatically annotate such tokens. Since their method labels all tokens with low confidence at once, there is no chance to re-estimate the probability. Therefore, we may waste some of the annotation effort.

In this paper, we propose a new active learning framework for sequence labelling. In the proposed algorithm, an informative token is selected by the system according to the marginal probability. The output probability of other informative tokens in the sequence are re-estimated by the system. After few iterations of annotation, the model becomes certain in predicting output of all tokens in the sequence. Thus, we need smaller amount of annotation cost than the cost when all informative tokens are labelled at once.

The rest of this paper is organized as follows. Section 2 discusses related works. Section 3 describes the Conditional Random Fields (CRFs) algorithm which we use as a classifier for our system. We propose our method in detail in section 4. Section 5 contains the experiment result and the discussion. Finally, we conclude our work and discuss the future work in Section 6.

2 Related Work

Settles et al. [3] had explored several fully-supervised active learning settings for sequence labelling. In contrast, our work is semi-supervised learning which requires fewer annotation effort compared to the supervised learning. Our work is mostly related to semi-supervised active learning proposed by Tomanek and Hahn in [2]. The main difference of our method from their method is the probability re-estimation. Since they annotate all informative tokens at once, there is no token with uncertain output left in the sequence.

Culotta and McCallum [4] introduced a system which can reduce a user effort on structured prediction tasks by probability re-estimation. An annotator is provided a list of labelling candidates generated from the system, and is asked to correct errors in a candidate starting from the least confident one. After each correction, the probability of the labelling is re-estimated. However, an annotator is required to verify all of the tokens in a candidate. In contrast to their method, we automatically decide the output for tokens with high confidence and only ask an annotator to label tokens with low confidence.

3 Conditional Random Fields (CRFs)

The objective of the sequence labelling task is to find an output label sequence $\mathbf{y} = (y_1, \dots, y_T) \in \mathbf{Y}$ of the input sequence $\mathbf{x} = (x_1, \dots, x_T) \in \mathbf{X}$. \mathbf{X} and \mathbf{Y} are

the sets of all possible input and output sequences, respectively. T is the length of a sequence. We will learn the mapping: $\mathbf{X} \rightarrow \mathbf{Y}$.

We adopt linear chain CRFs [5] which model the conditional probability of output label sequence \mathbf{y} given input sequence \mathbf{x} as

$$P_{\theta}(\mathbf{y}|\mathbf{x}) = \frac{e^{\theta \cdot \Phi(\mathbf{x}, \mathbf{y})}}{Z_{\theta, \mathbf{x}, \mathbf{Y}}}, \quad (1)$$

where $\Phi(\mathbf{x}, \mathbf{y}) : \mathbf{X} \times \mathbf{Y} \rightarrow \mathbb{R}^d$ is a function from a pair of input sequence \mathbf{x} and output sequence \mathbf{y} to a feature vector of d dimensions. $Z_{\theta, \mathbf{x}, \mathbf{Y}} = \sum_{\mathbf{y} \in \mathbf{Y}} e^{\theta \cdot \Phi(\mathbf{x}, \mathbf{y})}$, is the normalizing factor which can be computed efficiently using dynamic programming. $\theta \in \mathbb{R}^d$ is a set of model parameters learned from the labelled set by maximum likelihood estimation.

4 Active Learning for Sequence Labelling

In active learning, new sequences in each iteration are chosen by a query strategy. The query strategy returns either a sequence or a set of sequences which are likely to be the most informative sequences for training. Following Tomanek and Hahn in [2], we will regard the sequence \mathbf{x} with the lowest sequence probability as the most informative sequence. Then, we select a set of the most informative sequences from the unlabelled set in each iteration.

Subsequently, we divide tokens in the selected set into informative and uninformative tokens, based on the prediction confidence of the current model. We define the confidence measure in our work using the marginal probability computed as follows

$$P_{\theta}(y_j = y'|\mathbf{x}) = \frac{\alpha_j(y'|\mathbf{x}) \cdot \beta_j(y'|\mathbf{x})}{Z_{\theta}(\mathbf{x}, \mathbf{Y})}, \quad (2)$$

$\alpha_j(y'|\mathbf{x})$ is the forward score, which is the score of the prefix sub-sequence of \mathbf{x} to have the token at j annotated with y' . $\beta_j(y'|\mathbf{x})$ is the backward score, which is the score of the suffix sub-sequence of \mathbf{x} to have the token at j annotated with y' . Since our model is linear chain CRFs, α and β are computed using the algorithm similar to the forward-backward algorithm in standard hidden Markov models [5]. When the confidence of a token is less than the confidence threshold δ , we regard the token to be informative and a human annotator will annotate that token. Other tokens with high confidence are automatically annotated by the model. We iteratively annotate one token at a time starting from the least informative token, until there is no informative token left in the sequence.

Recall that a change in output probability in one token will affect the output probability of the other tokens in the same sequence. By labelling a token, the probability of that token is implicitly set to 1.0 while the probabilities of the other outputs of the same token are set to 0. We then re-estimate the probability of each output label after each manual annotation before any re-training. After re-estimation, if the system predicts a label of a token with the probability higher

than the threshold δ , we assume that they are correctly predicted. We employ the constrained Viterbi algorithm [4] for predicting output and estimating the output probability. The Viterbi decoding requires only few milliseconds and will not significantly affect the processing time of the whole system. Finally, we add the newly annotated sequences to the training set and start the next iteration. The learning will end after we have labelled all unannotated sequences.

5 Experiments

5.1 Data, Pre-processing, and Evaluation

We use the base NP chunking data from CoNLL-2000 shared task. The output labels are in IOB format [6]. Our feature set consists of unigram, bigram and trigram word and part-of-speech. We choose 50 longest sequences to be our initial set since long sequences are likely to contain more information than short sequences. The number of new sequences per iteration is fixed to 50 in all experiments.

Performance of each setting is evaluated by $F1$ versus the number of manually annotated tokens. $F1$ is measured following CoNLL evaluation [6]. The significance of $F1$ improvement is measured by McNemar's test.

5.2 Active Learning Settings

We employ CRFs described in section 3 as the labelling model in all settings. There are three baseline systems. The first baseline is Supervised-initial which is a supervised system using only the initial set as training data. The second baseline is the *Fully Supervised Active Learning* system (*FuSAL*). All tokens in each sequence are manually annotated. The last baseline is the *Semi-Supervised Active Learning* system (*SeSAL*) proposed by Tomanek and Hahn [2]. Firstly, all high confidence tokens which have the output probabilities exceed the confidence threshold δ , are automatically annotated by the current model. Subsequently, the low confidence tokens are manually annotated.

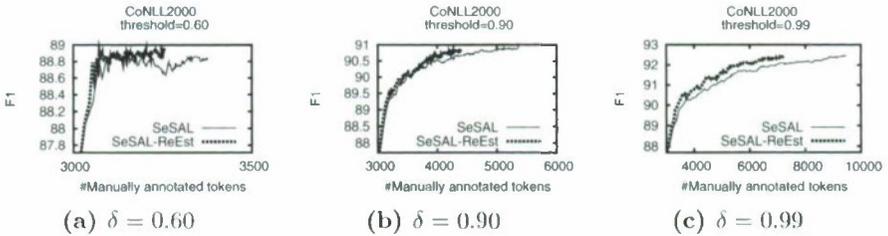
We propose the *Semi-Supervised Active Learning with Probability ReEstimation* system (*SeSAL-ReEst*). There are two main differences in *SeSAL* and *SeSAL-ReEst*. The first point is that a human annotator labels one informative token at a time in *SeSAL-ReEst* but label all informative tokens at once in *SeSAL*. The other point is that, we also re-estimate the probability after each annotation in *SeSAL-ReEst*.

5.3 Result

Fig. 1 shows that *SeSAL-ReEst* achieves similar $F1$ to *SeSAL* with less annotation cost. According to Table 1, we can reduce 3.61%, 18.01%, and 23.00% of annotation cost from *SeSal* when $\delta = 0.60, 0.90$ and $\delta = 0.99$, respectively. Table 1 also shows the number of mis-labelled tokens in the training data which is

Table 1. $F1$ using all sequences, number of manually annotated tokens, and number of mis-labelled tokens in the training set of each annotation setting

Settings- δ	without Re-estimation			with Re-estimation		
	$F1$	%Err _{train}	%Tag _{train}	$F1$	%Err _{train}	%Tag _{train}
Supervised-initial	87.71	(6.10)	1.43	-	-	-
<i>SeSAL</i> -0.60	88.84	4.72	1.60	88.96	4.73	1.54
<i>SeSAL</i> -0.90	90.96	3.58	2.54	90.78	3.64	2.08
<i>SeSAL</i> -0.99	92.45	2.38	4.48	92.45	2.51	3.42
<i>FuSAL</i>	93.86	0.00	100.00	-	-	-

**Fig. 1.** The number of manually annotated tokens and $F1$ using *SeSAL* and *SeSAL-ReEst* with confidence threshold $\delta = 0.60, 0.90, 0.99$

not significantly different in *SeSAL* and *SeSAL-ReEst*. Since the probability re-estimation increases the marginal probability of yet-unlabelled tokens to exceed the confidence threshold but produce quite similar output labels, *SeSAL-ReEst* requires less annotation cost than *SeSAL* but maintains the comparable $F1$.

With low confidence threshold, many erroneous tokens are not recovered and prevent the system from achieving high $F1$. Table 1 also shows the number of errors in the training set. With higher threshold, there are less errors in the training set thus we can achieve higher $F1$ than the setting with low threshold but with the higher cost of annotation effort.

6 Conclusion and Future Work

The semi-supervised active learning can reduce the human annotation cost by selectively labelling informative tokens. However, most of the informative tokens are already correctly predicted. The annotation and re-estimation will automatically annotate these tokens without any human effort. Hence, the proposed *SeSAL-ReEst* outperforms *SeSAL* in the terms of annotation cost to achieve a certain level of $F1$.

The processing time of probability re-estimation per iteration is only few milliseconds. However, the time consuming process is the CRFs training. Online learning which requires less time in model updating may be more appropriate

to the active learning task. We leave the improvement of the training algorithm to future work.

Moreover, even the system with high confidence setting, we cannot achieve the supervised $F1$ due to many errors in automatically labelled tokens. In other words, the current confidence measure does not succeed in selecting mis-labelled tokens. We have to re-design the query strategy in order to extract these mis-labelled tokens and have an annotator correct them.

Finally, we assume that the annotation difficulty of all tokens are the same. In a real scenario, some tokens may be harder to be labelled due to its ambiguity in the context. Our annotation cost should be re-defined to reflect the annotation difficulty.

Acknowledgements

We would like to thank your anonymous reviewers for valuable comments and suggestions. The first author is supported from Higher Educational Strategic Scholarships for Frontier Research Network, Thailand.

References

1. Cohn, D., Atlas, L., Ladner, R.: Improving generalization with active learning. *Machine Learning* 15(2), 201–221 (1994)
2. Tomanek, K., Hahn, U.: Semi-supervised active learning for sequence labeling. In: *ACL-IJCNLP 2007: Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, Suntec, Singapore, pp. 1039–1047. Association for Computational Linguistics (August 2009)
3. Settles, B., Craven, M.: An analysis of active learning strategies for sequence labeling tasks. In: *EMNLP 2008: Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Morristown, NJ, USA, pp. 1070–1079. Association for Computational Linguistics (2008)
4. Culotta, A., McCallum, A.: Reducing labeling effort for structured prediction tasks. In: *AAAI 2005: Proceedings of the 20th National Conference on Artificial Intelligence*, pp. 746–751. AAAI Press, Menlo Park (2005)
5. Lafferty, J.D., McCallum, A., Pereira, F.C.N.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: *ICML 2001: Proceedings of the Eighteenth International Conference on Machine Learning*, pp. 282–289. Morgan Kaufmann Publishers Inc., San Francisco (2001)
6. Tjong Kim Sang, E.F., Buchholz, S.: Introduction to the conll-2000 shared task: chunking. In: *CoNLL 2000: Proceedings of the 2nd Workshop on Learning Language in Logic and the 4th Conference on Computational Natural Language Learning*, Morristown, NJ, USA, pp. 127–132. Association for Computational Linguistics (2000)

Locally Centralizing Samples for Nearest Neighbors

Guihua Wen, Si Wen, Jun Wen, and Lijun Jiang

¹ South China University of Technology, Guangzhou 510641, China
crghwen@scut.edu.cn

² Hubei Institute for Nationalities, Ensi, China
wenj_64@163.com

Abstract. The k nearest neighbors classifier is simple and often results in good performance in problems. However, it can not work well on noisy and high dimensional data, as the structure composed of selected nearest neighbors on these data is easily deformed and perceptually unstable. This paper presents a locally centralizing samples approach with kernel techniques to preprocess the data. It creates a new sample for each original sample through its neighborhood and then replace it to be candidate for nearest neighbors. This approach can be justified by gestalt psychology and applied to provide better quality data for classifiers, even if the original data is noisy and high dimensional. The conducted experiments on challenging benchmark data sets validate the proposed approach.

1 Introduction

It empirically studied that k -nearest neighbors (KNN) classifier is simple and often results in good classification performance[1], so that its all kinds of variants have been proposed, such as new measures designed to select the optimal nearest neighbors[1,2] and local mean classifiers(LMC) proposed to resisting outliers [3,4]. However, they heavily depend on the collection of selected neighbors. The selected nearest neighbors on data with the sparse, noisy, or imbalanced property are easily deformed[10], which in turn leads to the worse performance[3]. This indicates that these classifiers are usually dependant on the quality of the data that they operate on, so that data preprocessing is necessary to remove the noise and to fill in the missing values. Generally eliminating the noisy samples is a hard problem if without any knowledge of data distribution. This paper proposes a locally centralizing samples (LCS) approach to modify the noisy data to normal data instead of removing them, which is then applied to design enhanced classifiers.

2 Locally Centralizing Samples Approach

All existing approaches to finding nearest neighbors heavily depend on some carefully selected measures[1,2]. However, when the training data is noisy or sparse, the selected neighbors by these measures are often conflict with human perception. In such case, the formed geometry shape composed of these selected neighbors is easily unstable, shown as Fig.1(B). When humans process visual stimuli,

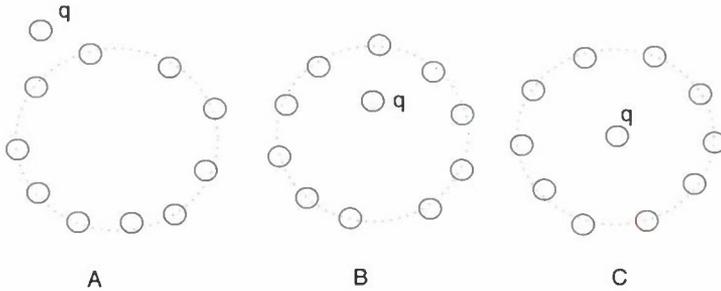


Fig. 1. Principle of visual perceptual laws (A) q is regarded as independent object of its nearest neighbors (B) q is taken as the part of the whole graph, but it is not perceptual stable as it is not at the center of graph. (C) q is taken as the part of the whole graph visually with robust stability.

global information often takes precedence over local information[8]. This means that we should measure the point not only by itself but by its neighborhood. Humans routinely classify others according to both their individual attributes and membership in higher order groups, so that individual attributes may be influenced and regulated by their group[7]. Generally noisy data is only small part of large data so that they can be revised to normal data by those normal data. According to Gestalt psychology[6], symmetry is an imprecise sense of harmonious and balance such that it reflects beauty or perfection. Central symmetry means that a geometric figure is called a symmetrical relatively a center, if all points are around the center point. We use this idea to locally centralizing samples through its nearest neighbors, and then replace its original one to be candidate for nearest neighbors. In this way, the selected nearest neighbors from locally centralized samples can be more consistent with our perceptual law, so that the classification can be performed better. This can be illustrated by Fig.1, where graph B is not stable and we intend to move the query q to the center of formed neighbor graph to remain the stability as the graph C shown. Now we give an algorithm to implement the LCS in the context of classification from statistics using Euclidean distance[5], denoted as ELCS.

ELCS(X, ξ, r)

/* X be training samples and $\xi(x_i)$ denotes the class of the sample x_i in X , r be the size for locally centralizing samples*/

Step 1. Select an sample from X , denoted as q

Step 2. Apply Euclidean distance d_e to find r nearest neighbors for q with the same class label, denoted as $\Omega(q, d_e, r)$

$$\Omega(q, d_e, r) = \{x_{\sigma(i)} \in X | d_e(q, x_{\sigma(i)}) \leq d_e(q, x_{\sigma(i+1)}), 1 \leq i \leq k\}$$

where σ be the permutation of index of samples in X , $d_e(q, x_{\sigma(1)}) \leq d_e(q, x_i)$, and $x_i \in X$.

Step 3. Generate the new sample for q by

$$q^b = \frac{1}{r + 1} \sum x \in \Omega(q, d_e, r)$$

Step 4. Repeat above all steps till all new samples are generated.

ELCS can not work well on the nonlinear data. This can be solved by using kernel function $k_f(x, y)$ to define the kernel distance $d_h(x, y)$ [9]:

$$d_h(x, y, k_f) = \sqrt{k_f(x, x) - 2k_f(x, y) + k_f(y, y)}$$

In this way, d_h can be applied to define the neighborhood for locally centralizing samples. This approach is called kernel locally centralizing samples(**KLCS**) approach.

3 Designed New Classifiers

Theoretically LCS acts as a smoother of the distribution of training samples, independent of classifiers used. Here we apply them to KNN and LMC classifiers, where LCS is only for training samples while the query sample keeps unchanged as its class label is not available.

ELCS-KNN(q, X, ξ, r, k)

/* X is the training sample set and $\xi(x_i)$ denotes the class of the sample x_i in X , r be the size for ELCS and k be the neighborhood size for classification*/

Step 1. Generate new samples from X by ELCS, denoted as X^b .

Step 2. Find k nearest neighbors for q from X^b using Euclidean distance, denoted as $\Omega(q, k)$

Step 4. Classify q into class ω_j if

$$\omega_j = \arg \max_{j \in \{1, 2, \dots, N_c\}} \{n_j = |\{x_i : x_i \in \Omega(q, k) \wedge \xi(x_i) = \omega_j\}|\}$$

where ω_j is the j th class, N_c is the number of total classes, and $|\cdot|$ is the cardinality of the set.

KLCS-KNN(q, X, ξ, r, k)

This approach is the same as ELCS-KNN except that it generates new samples from X by KLCS instead of by ELCS.

ELCS-LMC(q, X, ξ, r, k)

/* X is the training sample set and $\xi(x_i)$ denotes the class of the sample x_i in X , r be the size for ELCS and k be the neighborhood size for classification*/

Step 1. Generate new samples from X by ELCS, denoted as X^b .

Step 2. Select k nearest neighbors for q from $X^i \subseteq X^b$, denoted as $\Omega(q, k, \omega_i)$, where X^i is the training sample subset from class ω_i

Step 3. Compute the local mean vector, y_i , using k nearest neighbors:

$$y_i = \frac{1}{k} \sum x \in \Omega(q, k, \omega_i)$$

Step 4. Classify q into class ω_i if

$$\omega_i = \arg \min_i \{|q - y_i|\}$$

KLCS-LMC(q, X, ξ, r, k)

This approach is the same as ELCS-LMC except that it generates new samples from X by KLCS instead of by ELCS.

4 Experimental Results

4.1 Experimental Setup

In order to validate LCS approaches, we conducted extensive experiments by classifiers on benchmark artificial data and real data. The error rate is taken as the measure of performance of all compared classifiers[5,3,4]. In experiments, k takes the value over the range of [3, 6, ..., 30], and the parameter r for LCS takes the value from {1, ..., 9}. Kernel function type are tried among *linear, poly, rbf, and sigmoid* kernel functions, while the kernel parameters are taken from {0.1, ..., 0.9, 1, ..., 9}. When classifying, each data set is divided into training set and testing set according to the 'ModApte' split[11]. Ten such partitions are generated randomly for the experiments. On each partition, the compared classifiers are trained and tested for each pair of parameters, respectively, and then the best performance is reported.

4.2 On Artificial Data Sets

Using artificial data, we can control the number of the available samples and add noise according to the experimental purpose. To compare six classifiers in noise case, we perform the experiments on two spiral pattern data[13] and ring norm data set[12] with 200 points by adding random Gaussian noise to them where the mean of the noise is 0 and the variance is 0.0, 0.05, 0.1, ..., 0.45 respectively. It can be observed from Table.1 that on two noisy data, the classifiers enhanced by LCS performs obviously much better than the original ones does in terms of the average accuracy and standard deviation. This means that the classifiers with LCS is stronger to resisting in noise disturbance. To validate the better ability of the proposed LCS to deal with high dimensional data, we do experiments on ring norm data set[12] and p -dimensional norm data[4], as they can be generated by using different dimensions. It can be observed from Table.1 that the classifiers with LCS is more robust to the dimensionality and shows a favorable behavior in high dimensions.

Table 1. Accuracies of classifiers on noisy and high dimensional artificial data set(%)

Data	KNN	ELCS-KNN	KLCS-KNN	LMC	ELCS-LMC	KLCS-LMC
spiral (noise)	82.03±15.00	84.25±13.35	85.08±12.17	79.98±14.36	83.52±12.92	84.03±12.27
ring(noise)	59.14± 1.33	80.12± 2.64	85.10±2.42	83.36± 4.06	84.78± 3.66	86.58± 3.33
p-norm(dim)	55.53± 8.59	79.69± 2.88	81.67±3.00	83.61± 4.62	85.06± 3.91	87.03± 2.76
ring(dim)	44.91±24.97	88.90± 2.99	92.12± 2.23	92.17± 4.06	93.62± 3.76	94.96± 3.65

4.3 Experiments on Real Data Sets

To be practical, we also perform experiments on benchmark real data sets from UCI Repository of machine learning databases[15], where the records with missing values and non-numeric attribute are all removed. It can be observed from Table 2 that classifiers enhanced by LCS obviously outperforms KNN and LMC on average accuracy. These results do indicate the significant value of the proposed idea and the classifier. This also reenforces the idea ever justified by relative transformation[14] that Gestalt laws can be geometrically modeled and then applied to perform the classification better.

Table 2. Accuracies of six classifiers on real data sets (%)

Data	KNN	ELCS-KNN	KLCS-KNN	LMC	ELCS-LMC	KLCS-LMC
wine	76.35±5.21	78.85±5.21	80.58±5.40	78.27±5.13	80.96±6.04	82.12±5.59
dermatology	89.15±2.60	91.42±2.37	91.51±2.31	93.30±2.57	93.40±2.52	93.58±2.59
diabetes	75.35±2.11	76.09±2.18	77.13±1.25	74.96±2.04	76.26±2.04	77.04±1.61
ionosphere	86.06±1.83	93.56±2.36	94.23±2.22	91.06±3.01	93.65±2.32	95.58±2.09
glass	69.84±6.43	72.46±5.72	74.43±5.31	71.80±5.40	73.61±6.01	74.75±5.31
optdigits	98.75±0.33	98.97±0.31	98.97±0.31	99.10±0.47	99.21±0.41	99.21±0.41
segmentation	82.54±3.17	84.29±2.94	85.71±3.51	83.02±3.09	85.24±3.82	85.87±4.26
yeast	59.75±2.15	60.34±2.41	60.32±2.45	58.78±2.33	59.64±2.28	59.66±2.12
yaleface	65.33±3.18	71.56±4.16	72.89±3.28	66.44±4.12	70.44±4.57	72.67±3.48
iris	97.33±0.94	98.44±1.83	99.78±0.70	97.33±2.04	97.78±1.48	99.56±0.94
avg	80.04± 2.79	82.59± 2.94	83.55±2.67	81.40± 3.02	83.01±3.14	84.00±2.78

5 Conclusion and Future Work

This paper presents a locally centralizing samples approach that can effectively modify the noise data to norm data instead of removing them. This approach also makes the boundary of classes more separable so that the imbalanced problem can be solved. This approach is justified by gestalt psychology which means the formed geometry of data should be regular and symmetry as good as possible[6]. One of its implementation ways is called bootstrap approach from statistics[5]. However, this approach is only applied to design nearest neighbor classifier instead of k nearest neighbors classifier. We applied LCS to design several classifiers which can work better even if the original data is noisy or high dimensional. In the future, a lot of techniques will be applied to prompt LCS and then applied to the advanced classifiers such as support vector machine.

Acknowledgments

This work was supported by China National Science Foundation under Grants 60973083, "the Fundamental Research Funds for the Central Universities, SCUT", Guangdong Science and Technology project under Grants 2007B030803006.

References

1. Tristan, M.H., Stephane, R.: Tailored Aggregation for Classification. *IEEE Trans. Pattern Anal. Mach. Intell.* 31, 2098 (2009)
2. Wang, H.: Nearest neighbors by neighborhood counting. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28, 942 (2006)
3. Mitani, Y., Hamamoto, Y.: A local mean-based nonparametric classifier. *Pattern Recognition Letters* 27, 1151 (2006)
4. Li, B., Chen, Y.W., Chen, Y.Q.: The Nearest Neighbor Algorithm of Local Probability Centers. *IEEE Trans. Syst., Man, Cybern* 38, 141 (2008)
5. Hamamoto, Y., Uchimura, S., Tomita, S.: A bootstrap technique for nearest neighbor classifier design. *IEEE Trans. Pattern Anal. Mach. Intell.* 19, 73 (1997)
6. Desolneux, A., Moisan, L., Morel, J.: Computational gestalts and perception thresholds. *Journal of Physiology - Paris* 97, 311 (2003)
7. Bergman, T.J., et al.: Hierarchical Classification by Rank and Kinship in Baboons. *Science* 302, 1234 (2003)
8. Goto, K., Wills, A.J., Lea, S.E.G.: Global-feature classification can be acquired more rapidly than local-feature classification in both humans and pigeons. *Animal Cognition* 7 (2004)
9. Peng, J., Heisterkamp, D.R., Dai, H.K.: Adaptive Quasiconformal Kernel Nearest Neighbor Classification. *IEEE Trans. Pattern Analysis and Machine Intelligence* 26(5), 656–661 (2004)
10. Wen, G., Jiang, L., Wen, J.: Using Locally Estimated Geodesic Distance to Optimize Neighborhood Graph for Isometric Data Embedding. *Pattern Recognition* 41, 2226 (2008)
11. Lam, W., Han, Y.: Automatic Textual Document Categorization Based on Generalized Instance Sets and a Metamodel. *IEEE Trans. Pattern Anal. Mach. Intell.* 25, 628 (2003)
12. Breiman, L.: Arcing classifiers. *Ann. Statist.* 26, 801 (1998)
13. Singh, S.: 2D spiral pattern recognition with possibilistic measure. *Pattern Recognition Lett.* 19, 131 (1998)
14. Wen, G., et al.: Local relative transformation with application to isometric embedding. *Pattern Recognition Letters* 30, 203 (2009)
15. Asuncion, A., Newman, D.J.: UCI Machine Learning Repository. University of California, School of Information and Computer Science, Irvine, CA (2007). <http://www.ics.uci.edu/~mllearn/MLRepository.html>

Gait Planning Research for Biped Robot with Heterogeneous Legs*

Jun Xiao¹, Xing Song^{1,2}, Jie Su¹, and Xinhe Xu¹

¹ College of Information Science and Engineering, Northeastern University,
Shenyang, China 110004
xiaojun@ise.neu.edu.cn

² State Key Laboratory of Robotics, Shenyang Institute of Automation,
Chinese Academy of Sciences, Shenyang, China 110016

Abstract. Biped Robot with Heterogeneous Legs (BRHL) is a novel robot model, which consists of an artificial leg and an intelligent bionic leg. The artificial leg is used to simulate the amputee's healthy leg and the intelligent bionic leg works as the intelligent artificial limb. This paper discusses how a BRHL robot imitates a person's walking from the points of gait identification, gait generation and gait control. Simulative and practical system experiments prove the validity of the presented plan and proposed algorithm. This robot's design provides an excellent platform for the research of intelligent prosthetic leg.

Keywords: Intelligent bionic leg; Gait planning; Intelligent prosthesis.

1 Introduction

Intelligent bionic leg is used to replace the malformed limb of amputee in the domain of healing biomedicine. Research of intelligent prosthesis needs a lot of various experiments, but the amputee can't afford so many repeated experiments, so the progress of intelligent prosthesis is undoubtedly affected. The proposed biped Robot with Heterogeneous Legs (BRHL) [1] consists of an artificial leg and an intelligent bionic leg, as shown in Fig. 1(a). The artificial leg is used to simulate the amputee's healthy leg and the intelligent bionic leg works as the intelligent artificial limb.

The artificial leg has six Degrees of Freedom (DOFs), the joints are active joints driven by motors and linked with rigid body. The knee joint has multi-bar closed-chain structure. It is a semi-active joint. Biped robot is a natural unstable system. In order to simulate the situation that amputees walk in line with intelligent prosthesis(IP) dynamically, an assistant quadricycle system is designed to keep the robot walking stably. The whole BRHL system is shown in Fig. 1(b).

This paper discusses how a BRHL robot imitates a person's walking from the points of gait identification, gait generation and gait control. Section 2 describes

* This work is supported by Chinese National Programs for High Technology Research and Development.2005AA420230.



Fig. 1. (a) Simplified BRHL virtual prototype (b) BRHL experiment system

gait identification and planning of bionic leg. Gait simulation as well as united control simulation of BRHL is conducted in Section 3. Based on these simulating experiments, practical BRHL prototype is built and experiment results are depicted in Section 4. Conclusion and prospects are drawn in Section 5.

2 Gait Identification and Planning of Bionic Leg

The common methods of biped robot's gait planning includes the method based on gait data of human body[2]; the methods based on the calculation of dynamics and kinematics [3][4]; the method based on the artificial neural network and genetic algorithm[5] and the methods based on Central Pattern Generator [6][7].

Compared with the common biped walking robot, according to the BRHL's characteristics, artificial leg's gait is obtained by leg gait planning artificially and bionic leg's gait is designed to follow the artificial leg's motion.

2.1 Gait Identification with Process Neural Network

Gaits will have big differences in different terrains, and each joint provides different torque. Five terrains of flat, up-slope,down-slope,upstairs and downstairs are chosen here.

Ground Reaction Force (GRF) in different terrains is used for gait identification [8]. 6D force sensor in ankle joint of intelligent bionic leg is used to measure three forces and three torques from three directions. Then suitable gait data are looked for from gait data base according to the terrain, which is used to control damper output force of bionic leg knee joint to follow artificial leg. If two leg information is not symmetry, artificial leg regulation is used.

Process neural network is adopted for gait identification. Output layer of process neural networks completes space weight congregation of latent signals and time congregation computation. Suppose $\{b_i(t)\}$ are a group of base functions of process neural networks input space $C[0, T]$, then weight functions could be expressed as limited term combination of the base functions.

Suppose system input is: $X(t) = (x_1(t), x_2(t), \dots, x_n(t))$

Then system output is:

$$y = \sum_{i=1}^m v_i f \left(\sum_{j=1}^n \int_0^t \left(\sum_{l=1}^L w_{ji}^l b_l(t) \right) x_j(t) dt - \theta_i \right) \tag{1}$$

$$= \sum_{i=1}^m v_i f \left(\int_0^t \sum_{j=1}^n (\bar{w}_{ji} x_j(t)) dt - \theta_i \right) \tag{2}$$

Where

$$\bar{w}_{ji} = \sum_{l=1}^L w_{ji}^l b_l(t) \tag{3}$$

Network error function is:

$$E = \sum_{k=1}^K (y_k - d_k)^2 \tag{4}$$

$$= \sum_{k=1}^K \left(\sum_{i=1}^m v_i f \left(\int_0^t \sum_{j=1}^n (\bar{w}_{ji} x_{kj}(t)) dt - \theta_i \right) - d_k \right)^2 \tag{5}$$

With gradation descending method, network weight study rules are:

$$v_i = v_i + \alpha \Delta v_i \tag{6}$$

$$\bar{w}_{ji} = \bar{w}_{ji} + \beta \Delta \bar{w}_{ji} \tag{7}$$

$$\theta_i = \theta_i + \gamma \Delta \theta_i \tag{8}$$

Where, v_i is the connecting weight between latent layer and output layer, \bar{w}_{ji} is the connecting weight between node j of input layer and node i of latent layer, θ_i is the output threshold value of latent layer, d_k is the input sample k of desired output, α, β, γ are study efficiencies. $b_l(t)$ is base function, n is input node number, m is latent node number, K is the division number in $[0, T]$, L is base function number.

2.2 Gait Planning of Bionic Leg

The hip joint's motion track of bionic leg can be solved directly by that of artificial leg:

$$\theta_2^b(t) = \theta_2^a(t) + \frac{T}{2} \tag{9}$$

The knee joint's ideal motion track of bionic leg can also be solved by that of artificial leg directly:

$$ref\theta_3^b(t) = \theta_3^a(t) + \frac{T}{2} \tag{10}$$

Because control system could only provide limited driving force, the actual controlling input force/force moment has some constraints in the controlled object

model, especially in dynamic model. Therefore, the state space track the system can realize isn't a whole phase space and only a subset. If the required track δ belongs to the attainable track space Ω , that is $\delta \in \Omega$, then the ideal control law can be obtained. Else a optimized control curve exists, which can make the practical track most close to the required track.

Because MR damper of bionic leg is a limited driving, it may not realize an ideal motion track $ref\theta_3^b(t)$. To solve the optimized control law $U^*(t)$, $U^*(t) \in U_{ad}$ (U_{ad} is the allowed control set), and make the knee joint motion $\theta_3^b(t)$ of the bionic leg follow $ref\theta_3^b(t)$, the quadric optimized performance index function is:

$$\min J(U) = \int_0^1 (\delta\theta^T \delta\theta + \delta\dot{\theta}^T \delta\dot{\theta})dt \tag{11}$$

In it,

$$\delta\theta = ref\theta_3^b(t) - \theta_3^b(t) \tag{12}$$

$$\delta\dot{\theta} = ref\dot{\theta}_3^b(t) - \dot{\theta}_3^b(t) \tag{13}$$

The optimized control vector is

$$U = (T_3^b, I)^T \tag{14}$$

where T_3^b is the control torque of bionic leg's knee joint and I is the control current of damper.

The damper force F provided by damper is related with $\theta_3^b, \dot{\theta}_3^b$ and input current of damper. The constraint relationship is

$$F = f(\theta_3^b(t), \dot{\theta}_3^b(t), I) \tag{15}$$

In addition, there are the initial condition constraints,

$$\delta\theta(t_0) = 0, \delta\dot{\theta}(t_0) = 0 \tag{16}$$

The damper current constraint is:

$$0 \leq I \leq 2 \tag{17}$$

In the practical calculation, discretion of the continuous system is needed.

$$\min J(U(\cdot)) = \sum_{i=1}^n (\delta\theta(i)^T \delta\theta(i) + \delta\dot{\theta}(i)^T \delta\dot{\theta}(i)) \tag{18}$$

$$U = (T_3^b(i), I(i))^T \tag{19}$$

$$\delta\theta(i) = ref\theta_3^b(i) - \theta_3^b(i) \tag{20}$$

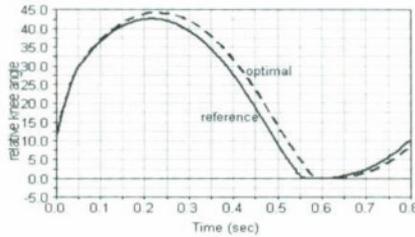
$$\delta\dot{\theta} = ref\dot{\theta}_3^b(i) - \dot{\theta}_3^b(i) \tag{21}$$

The solved $\theta^*(t)$ corresponding to $U^*(t)$ is called the optimized track or the extremal curve.

Table 1. Sample data in five terrains

Percent	GRF from vertical direction (KN)					
	upstairs	up-slope	downstairs	down-slope	flat	
number	1	0.0555	0.0453	0.0616	0.0497	0.0530
	2	0.1043	0.0878	0.1183	0.0953	0.1002
	3	0.1482	0.1286	0.1714	0.1379	0.1433
	4	0.1890	0.1692	0.2225	0.1794	0.1840
	5	0.2285	0.2101	0.2713	0.2198	0.2239
	6	0.2686	0.2531	0.3206	0.2616	0.2647
	7	0.3111	0.2994	0.3711	0.3059	0.3082ZE
	8	0.30821	0.3500	0.4240	0.3540	0.3562
	9	0.4102	0.4055	0.4800	0.4066	0.4090

99	0.0207	0.0199	0.0216	0.0203	0.0205	
100	0.0080	0.0080	0.0080	0.0080	0.0080	

**Fig. 2.** Gait Tracking Optimization of Knee Joint

3 Gait Identification and Planning of Bionic Leg

6D force sensor in ankle joint of intelligent bionic leg can measure the forces of axis x , y and z , as well as three torques (M_x , M_y , M_z). Table 1 is sample data in five terrains.

The simulation example of the optimized gait following is shown in Fig. 2

The dashed line stands for the ideal track of bionic leg's knee joint and the real line stands for the result of gait following.

4 Implementation of Practical System Experiments

To validate bionic leg's control scheme of knee joint and gait's humanoid performance of swinging phase, swinging and walking experiments are conducted in condition of planned gait. The motion track of knee joint of artificial leg and intelligent bionic leg is shown in Fig. 3(a)

It could be seen that there are many inflexion points in artificial curve. And the track of knee joint of intelligent bionic leg is smooth because of damper on it. And the practical experiment result of knee joint can only partly follow the ideal gait.

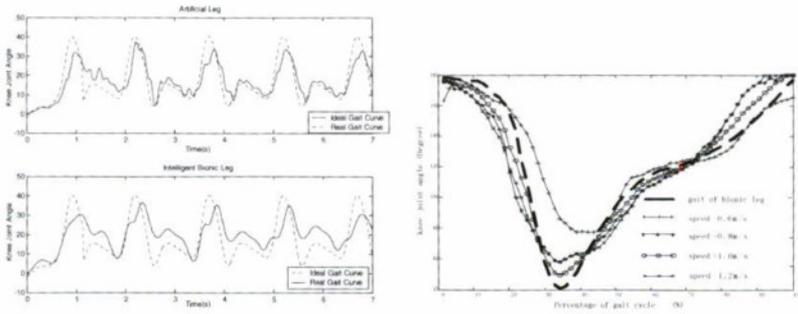


Fig. 3. (a) The Track of knee joint of artificial leg and intelligent bionic leg; (b) Swinging phase experiments of knee joint of intelligent bionic leg

Swinging phase experiments is shown in Fig. 3(b). The result indicates that there is still big error between practical gait and desired gait.

5 Conclusion

BRHL is an integration of common biped robot and intelligent prosthesis. It can well simulate the situation that human walks with IP. United simulation of two leg walking gait and swinging phase of artificial leg indicates that the simulation platform approaches to the practical system. Practical system of BRHL is built and practical experiments of swinging phase and walking are conducted. The practical control experiment results are presented.

References

1. Xu, X., Wang, B., Tan, J.: A new robot mode-biped robot with heterogeneous legs. *High Technique Communication* 12, 38–41 (2004)
2. Dasgupta, A., Nakamura, Y.: Making feasible walking motion of humanoid robots from human motioncapture data. In: *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1044–1049 (1999)
3. Nagasaka, K., Inoue, H., Inaba, M.: Dynamic walking pattern generation for a humanoid robot based on optimal gradient method. In: *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics* (1999)
4. Hualong, X.: Development and harmony control research of biped robot with heterogeneous legs. Northeastern University, Shenyang (2006)
5. Kun, A.L., Miller, W.T.: Control of variable speed gaits for a biped robot. *Robotics and Automation* 6, 19–29 (1999)
6. Williamson, M.M.: Robot arm control exploiting natural dynamics. Massachusetts Institute Technology, Cambridge (1999)
7. Selverston, A.I., Rowat, P.F.: Modeling phase synchronization in a biologically-based network. *Neural Networks, IJCNN* (1992)
8. Avci, E., Turkoglu, I.: Intelligent target recognition based on wavelet packet neural network. *Expert Systems with Applications* 29, 175–182 (2005)

Computer-Aided Diagnosis of Alzheimer's Disease Using Multiple Features with Artificial Neural Network

Shih-Ting Yang¹, Jiann-Der Lcc^{1,*}, Chung-Hsien Huang¹, Jiun-Jie Wang²,
Wen-Chuin Hsu³, and Yau-Yau Wai³

¹ Department of Electrical Engineering, Chang Gung University, Taiwan 333
jdlcc@mail.cgu.edu.tw

² Department of Medical Imaging and Radiological Sciences, Chang Gung University,
Taiwan 333

³ Department of Neuroscience, Chang Gung Memorial Hospital, Taiwan 333

Abstract. Alzheimer's disease (AD) is a progressively neuro-degenerative disorder. In the AD-related research, the volumetric analysis of hippocampus is the most extensive study. However, the segmentation and identification of the hippocampus are highly complicated and time-consuming. Therefore, a MRI-based classification framework is proposed to differentiate between AD's patients and normal individuals. First, volumetric features and shape features were extracted from MRI data. Afterward, Principle component analysis (PCA) was utilized to decrease the dimensions of feature space. Finally, a Back-propagation artificial neural network (ANN) classifier was trained for AD classification. With the proposed framework, the classification accuracy is reached to 88.27% by only using volumetric features and shape features. And, the result achieved up to 92.17% by using volumetric features and shape features with the PCA.

Keywords: Alzheimer's disease, magnetic resonance imaging, shape descriptors, Artificial Neural Network, Principle component analysis.

1 Introduction

Alzheimer's disease (AD) is a progressively neuro-degenerative disorder. Up to present, AD affects approximately 26 million people worldwide, and this number may increase fourfold by 2050.

Diagnostic criteria for AD are currently based on clinical and psychometric assessment. The main procedures for the evaluation of probable AD patients are neuropsychological tests. In clinical, magnetic resonance imaging (MRI) is a very important tool in diagnosing AD because it can qualitatively measure the neuronal loss by the shrinkage of the structures-of-interest more easily. Consequently, MRI has demonstrated that volumetric atrophy appears in the early stages of AD [1].

In addition, the enlargement of ventricles is also a significant characteristic of AD due to neuronal loss [2]. Ventricles are filled with cerebro-spinal fluid (CSF) and

* Corresponding author.

surrounded by gray matter (GM) and white matter (WM). As a result, by measuring the ventricular enlargement, hemispheric atrophy rate shows higher correlation with the disease progression when compared to the medial temporal lobe atrophy rates, and reveals significant variation between normal individuals and AD.

In this study, a MRI-based classification framework is proposed to distinguish AD's patients from normal individuals. Section 2 explains the proposed framework comprising system flowchart and selected shape features. Statistical analysis and experimental results are described in Section 3. Finally, the conclusion is included in Section 4.

2 Flow Chart and Feature Extraction

Figure 1 illustrates the flowchart of the proposed image-aided AD diagnosis system. Details of each step are explained in the following.

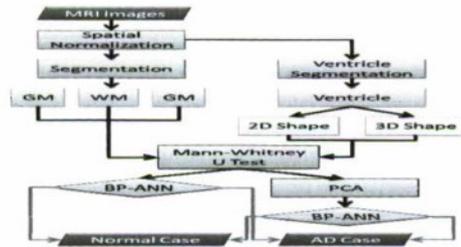


Fig. 1. Flowchart of the proposed image-aided AD diagnosis system

2.1 Spatial Normalization of MRI Data

Spatial normalization is a procedure to register a set of MRI data to a standard spatial coordinate system, also known as Talairach and Tournoux coordinate system [3]. Therefore, each voxel in the MRI data is compared with the voxel at the same position of other registered MRI data or reference-MRI template. In this study, all of the 3-D MRI sets were normalized to ICBM MRI template by using an optimum 12-parameter affine transformation and a Bayesian framework.

2.2 Volume Features

The volumes of GM, WM and CSF indicated important information, especially in brain degeneration diseases [4]. Hence, a clustering-based segmentation algorithm is adopted to extract GM, WM and CSF probability maps from the source MRI data. The value of each pixel in the corresponding probability map denotes the posterior of the pixel belonging to the tissue by giving its gray intensity. The volumes of GM, WM and CSF and whole brain are obtained by the following equations:

$$\text{volume}_{\text{GM}} \approx \sum_{i \in I} (P(C_{\text{gray}} | f(i)) > 0.5) \quad (1)$$

$$\text{volume}_{\text{WM}} \approx \sum_{\forall i \in I} (P(C_{\text{white}} | f(i)) > 0.5) \tag{2}$$

$$\text{volume}_{\text{CSF}} \approx \sum_{\forall i \in I} (P(C_{\text{CSF}} | f(i)) > 0.5) \tag{3}$$

$$\text{volume}_{\text{Whole}} \approx \sum_{\forall i \in I} (P(C_{\text{GM-WM}} | f(i)) > 0.5) \tag{4}$$

where i is any pixel of the MRI data and $f(i)$ stands for the gray level of i . Figure 2 illustrates the segmentation results of the normal individual and AD patient.

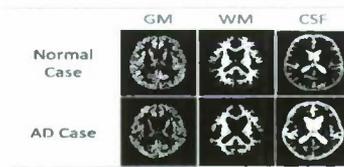


Fig. 2. Segmentation results of the normal individual and AD patient

Binary ventricle volume data, $M(x, y, z)$, are extracted from MR images using region growing algorithm and a threshold which was found through double threshold algorithm [5]. After the thresholding, the binary ventricle regions are obtained using the fill, erosion and dilation methods. The edges of binary images are detected by using the Sobel operator on a slice-by-slice basis. Then segmented region will construct a mask image, where 1 stands for the ventricle pixel in mask image and 0 stands for the non-ventricle pixel. Lastly, Eq. (5) is used to measure the cerebral ventricle, as shown in Figure 3 (a) and (b). Where i is any pixel of the mask data, M is mask image and $f(i)$ denotes for the gray level of i .

$$\text{volume}_{\text{Ventricle}} \approx \sum_{\forall i \in M} (P(C_{\text{Ventricle}} | f(i)) = 1) \tag{5}$$

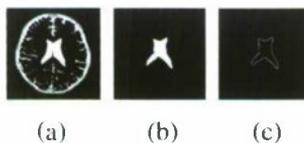


Fig. 3. (a) CSF binary map, (b) ventricle mask image, and (c) edge of ventricle mask image

2.3 Shape Features

In contrast to the volume features, which are extracted from the whole three-dimensional volume, the local shape features, such as area, distances between salient points and symmetry, are obtained from a single 2-Dimensional slice [6].

In the feature of 3-D shape, we use leave-one-out method to construct training set and testing set. Then we build up two sets of probability map using Eq. (6) and Eq. (8) for the normal and patients in training set, as shown in Figure 4 (a) and (b). Where

M is the number of normal controls, N is the number of AD patients and I represent the grey value of the ventricular mask image.

$$P_{Normal}(x, y, z) = \frac{1}{M} \sum_{i=1}^M I_{Normal}^i(x, y, z) \quad (6)$$

$$P_{AD}(x, y, z) = \frac{1}{N} \sum_{i=1}^N I_{AD}^i(x, y, z) \quad (7)$$

Following, we have the discriminate map by subtracting the normal probability map from the AD probability map, as shown in Figure 4 (c). Finally, matching coefficient (MC) and the discriminate map are calculated using Eq. (8). Here, $D(x, y, z)$ is the discriminate map and T stands for the testing ventricular mask image.

$$MC_{Normal/AD}^T = \sum_{x,y,z} D(x, y, z) T_{Normal/AD}^T(x, y, z) \quad (8)$$

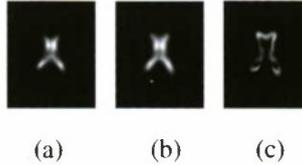


Fig. 4. (a) Probability of the normal controls, (b) probability of the AD patients and (c) discriminate map

In this approach, the 2-D shape features used, including (1) Area, (2) Perimeter, (3) Compactness, (4) Elongation, (5) Rectangularity, (6) Distances, (7) Minimum thickness, and (8) Mean signature value.

2.4 Back-Propagation Artificial Neural Network Architecture

In this approach, a three-layer BP-ANN is employed for classification task. The input layer contains 20 neurons, and the output layer has one neuron. Hidden layer is composed of 17 neurons [7]. The maximum iterations are set to 5000 epochs, and the output error of the validation is less than 0.01. The output value is within the range (0.0-1.0). A threshold (in our case, it is 0.5) is applied to classify each individual. If the output value is less than the threshold, the subject is assigned to probable AD group; on the other hand, the subject is denoted as normal control group. The neural classifier is trained 10 times to get reliable results. Thirty subjects (AD = 12, Normal = 18) are used in the training set randomly.

3 Experimental Results

3.1 Material

The whole dataset consists of two groups: 24 patients of probable Alzheimer's disease and 28 normal controls of comparable age. Twenty-eight individuals are normal

controls (18 males, 10 females), mean age was 67 ± 5.67 years, with education time of 10 ± 4.8 years. The average score of MMSE was 28 ± 1.24 . Twenty-four individuals were diagnosed as probable AD patients (11 male, 13 female), mean age was 71 ± 7.37 years, with education time of 6.96 ± 5.84 years. All patients were based on the MMSE complemented by verbal memory, figurative memory and visuospatial tests. The average score of MMSE was 14.38 ± 6.55 .

3.2 Statistical Analysis and Classification

Mann-Whitney U test was performed on each feature to evaluate its discriminative power. The p-values obtained from the test provide a generally known and comparable criterion. It rejects the null hypothesis of equal distributions when $p < 0.05$. Table 1 illustrates the statistical results of volume and shape features. In the experiment, the circularity and reectangularity are rejected ($p > 0.05$) in the following steps of classification.

Table 1. Statistical analysis of features

Features	Mean volume in [mm] \pm S.D.		
Volume	Normal	AD	p-value
V _{GM}	849.5 \pm 62.1	776.6 \pm 114.3	0.011
V _{WM}	621.6 \pm 57.3	534.5 \pm 71.9	0.014
V _{CSF}	849.6 \pm 137.1	969.8 \pm 117.8	0.038
Shape	Normal	AD	p-value
Area	1581.1 \pm 268.3	2206.4 \pm 713.8	0.013
Area (PR)	614.4 \pm 112.1	901.7 \pm 211.6	0.004
Area (PL)	611.7 \pm 118.4	907.9 \pm 234.1	0.001
Area (FR)	132.8 \pm 98.5	253.9 \pm 176.1	0.008
Area (FL)	140.5 \pm 76.9	276.4 \pm 191.0	0.007
Perimeter	214.3 \pm 18.9	283.8 \pm 36.3	0.013
Circularity	43.9 \pm 5.6	37.0 \pm 3.1	0.027
Elongation	1.2 \pm 0.7	1.3 \pm 0.1	0.022
Rectangularity	0.5 \pm 0.1	0.6 \pm 0.1	0.011
d(A,G)	34.7 \pm 3.1	39.8 \pm 6.4	0.004
d(B,G)	35.1 \pm 2.9	42.3 \pm 5.8	0.022
d(C,G)	37.3 \pm 2.1	42.6 \pm 5.1	0.026
d(D,G)	35.1 \pm 3.7	41.3 \pm 4.6	0.029
d(A,C)	73.2 \pm 5.1	82.4 \pm 12.9	0.016
d(B,D)	69.5 \pm 6.7	80.9 \pm 10.4	0.003
Min thickness	25.9 \pm 2.1	29.5 \pm 3.7	0.011
Mean Sig.	24.5 \pm 2.9	29.1 \pm 2.8	0.014

In fact, some of features may be redundant or have highly correlation. Therefore, PCA [8] was introduced to reduce the dimensions of the feature space. The principal components which contribute 95% to the total variation in data set were chosen herein. More specifically, to train a volume-feature-based classification, all the volume features were adopted. To train a shape-feature-based classification, only the first five principal components which convey a large amount of information quantified by

95% energy were adopted. In the case of using both shape and volume features, the first six principle components were employed. For the classification, BP-ANN was utilized to train a classifier.

Table 2 shows the accuracy, sensitivity, and specificity when using various features. Obviously, incorporating with shape features, volume features, and PCA shows excellent classification ability than others. The accuracy, sensitivity and specificity have been improved to 92.17%, 79.91% and 88.61%, respectively.

Table 2. Classification results

	Volume features	Shape features	Volume + Shape features	Volume + Shape features + PCA
Accuracy	76.03%	78.92%	88.27%	92.17%
Sensitivity	73.43%	80.47%	76.63%	79.91%
Specificity	78.69%	71.27%	87.31%	88.61%

4 Conclusions

In this study, we present a classification framework for image-aided diagnosis for AD by using easy-extractable volume and shape features. With the proposed framework, the classification accuracy is reached to 88.27% by only using volumetric features and shape features. Moreover, the correctness is up to 92.17% by using volumetric features and shape features with the aid of PCA. From the experimental results, it is implied that combining volume features and shape features to classify AD is achievable due to their low computational complexity and discriminate capability.

Acknowledgements

This work was supported by National Science Council, R. O. C. with Grant No. NSC98-2221-E-182-040-MY3 and Chang Gung Memorial Hospital with Grant No. CMRPD270051.

References

1. Vemuri, P., Wiste, H.J., Weigand, S.D., Shaw, L.M., Trojanowski, J.Q., Weiner, M.W., Knopman, D.S., Petersen, R.C., Jack Jr., C.R.: MRI and CSF biomarkers in normal, MCI, and AD subjects: predicting future clinical change. *Neurology* 73(4), 294–301 (2009)
2. Nestor, S., Rupsingh, R., Accomazzi, V., Borrie, M., Smith, M., Wells, J., Bartha, R.: Changes in brain ventricle volume associated with mild cognitive impairment and alzheimer disease in subjects participating in the alzheimer's disease neuroimaging initiative. *Alzheimer's and Dementia* 3, S114 (2007)
3. Talairach, J., Tournoux, P.: Co-Planar Stereotaxic Atlas of a Human Brain. In: Dimensional Proportional System: An Approach to Cerebral Imaging. Georg Thieme Verlag (1988)
4. Fritzsche, K.H., von Wangenheim, A., Abdala, D.D., Meinzer, H.P.: A computational method for the estimation of atrophic changes in Alzheimer's disease and mild cognitive impairment. *Computerized Medical Imaging and Graphics* 32, 294–303 (2008)

5. Otsu, N.: A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man and Cybernetics* 9, 62–66 (1979)
6. Wang, J., de Haan, G., Unay, D., Soldea, O., Ekin, A.: Voxel-based discriminant map classification on brain ventricles for Alzheimer's disease. In: *Medical Imaging*, vol. 7259 (2009)
7. de la Escalera, A., Moreno, L.E., Salichs, M.A., Armingol, J.M.: Road traffic sign detection and classification. *IEEE Transactions on Industrial Electronics* 44(6), 848–859 (1997)
8. Jolliffe, I.T.: *Principal Component Analysis*, 2nd edn. Springer Series in Statistics. Springer, New York (2002)

A Hierarchical Multiple Recognizer for Robust Speech Understanding

Takahiko Yokoyama, Kazutaka Shimada, and Tsutomu Endo

Department of Artificial Intelligence, Kyushu Institute of Technology
680-4 Iizuka Fukuoka 820-8502 Japan
{t_yokoyama, shimada, endo}@pluto.ai.kyutech.ac.jp

Abstract. In this paper, we propose a simple and effective method for speech understanding. The method incorporates some speech recognizers. We use two types of recognizers; a large vocabulary continuous speech recognizer and a domain-specific speech recognizer. The multiple recognizer is a robust and flexible method for speech understanding. Words in different utterances often contain relations. For example, users frequently input the parameter value after speaking command names to a system. We handle the relation by a hierarchical multiple recognizer. We compared the proposed method with a non-hierarchical method. Our method outperformed the non-hierarchical method.

Keywords: Multiple speech recognizer, Output selection, Hierarchical method.

1 Introduction

Speech understanding systems have been developed for practical use recently. One approach to develop speech understanding systems with higher accuracy is to construct a speech understanding method using keywords, key phrases, or sentence templates [1,2]. However keyword based methods contain a problem; misunderstanding of non-commands in a dialogue. Here assume that the word “Search” is a command for a system, and a user mutters “This is a search result that I got yesterday.” in front of a microphone of the system. In this case, keyword-based speech understanding methods often extract the word “search” in the mutter as the command for the system. Therefore, the speech understanding system needs to detect non-command utterances in a dialogue. Several utterance verification methods have been proposed [3,4,5].

In addition, words in different utterances often contain relations. For example, users frequently input the parameter value after speaking command names to a system. Lane et. al. [6] have reported a hierarchical topic classification method for speech recognition. They used the relation for the hierarchical recognizer. The method switched the language model in the recognizer on the basis of the current topic in a dialogue.

In this paper we use the speech understanding method proposed by [5] as the basic approach. The task of the speech understanding is an image edit and

management application. We also apply a hierarchical approach to the speech understanding method. For the task, we compare the proposed method with a non-hierarchical method.

2 OGSS: A Multiple Speech Recognizer

In this section, we describe the basic idea of our multiple recognizer. It is based on a large vocabulary continuous speech recognizer (LVCSR) and some domain-specific speech recognizers (DSSR) [5]. We called it "One Generalist and Some Specialists (OGSS) model". In our system, the LVCSR is the generalist, namely domain-independent, and the DSSRs are specialists, namely domain-dependent. Here we use one LVCSR for non-command utterances and some DSSRs for command utterances. By using this method, we can distinguish commands from a chat (non-command utterances).

In this process, we focus on a difference of outputs generated from each recognizer. If an input is a command utterance, a DSSR and the LVCSR generate similar outputs on phoneme-level because the LVCSR is domain independent. On the other hand, if the input is not a command utterance, they often generate different outputs even on the phoneme-level because all the DSSRs never generate the correct result for non-command utterances. In our method, we compute the edit distance of phonemes of utterance-level and word-level by using a DP matching algorithm.

The rules to judge an utterance are applied in the following order:

1. Compute the edit distance of the utterance-level (ED_{utter}) between the LVCSR and each DSSR. For the outputs of which the edit distance is less than $thresh_{utter}$, we select the output of the DSSR which contains the minimum ED_{utter} as the final output.
2. Compute the edit distance of the word-level (ED_{word}) between the LVCSR and each DSSR. For the output of which the edit distance is less than $thresh_{word}$, we select the output of the DSSR which contains the minimum ED_{word} as the final output. Otherwise, the LVCSR as the final output.

The ED_{utter} is the edit distance value on the utterance-level. The ED_{word} is the average of the edit distance value computed on word-level. These values are normalized by the number of phonemes in the outputs. The $thresh_{utter}$ and $thresh_{word}$ are threshold values for the judgment. These values are based on the previous work [5]. In the paper, $thresh_{utter} = 0.26$ and $thresh_{word} = 0.08$. Figure 1 shows examples of this process. See [5] for more details.

3 Hierarchical Method

Words in different utterances often contain a dependency relation. For example, users frequently input the parameter value after speaking command names to a system. We treat the relation by a hierarchical multiple recognizer. In this section, we describe a hierarchical method for the OGSS model speech recognizer.

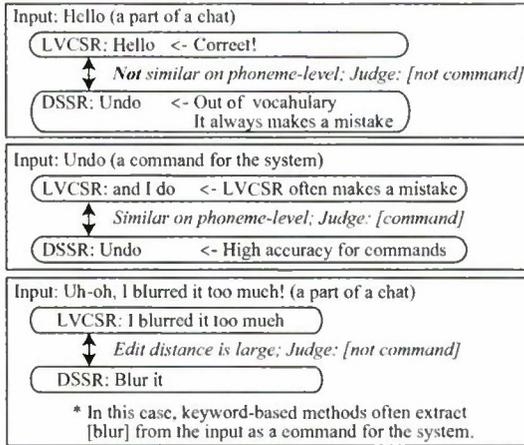


Fig. 1. Examples of the utterance verification

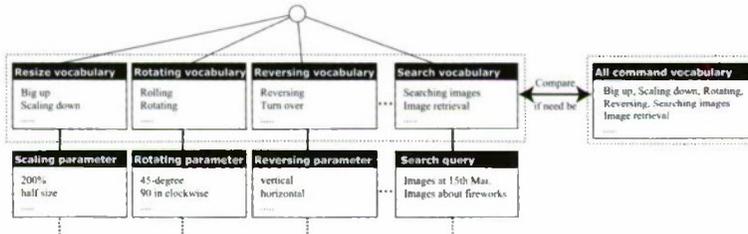


Fig. 2. The hierarchical method

Figure 2 shows an example of the hierarchical method. A rectangle in the figure denotes a speech recognizer. The system consists of some DSSRs that are segmented by each command category and a DSSR with all command vocabularies. First, it selects the output with the minimum edit distance from the segmentalized DSSRs. Here, we apply a threshold to the output. If the output from the segmentalized DSSR contains high confidence, we select the output as the final result of the hierarchical method. We regard the edit distance value in the OGSS model as the confidence measure for the process. The threshold is applied to the edit distance of the utterance-level. The threshold $thresh_{comb}$ is 0.14 in this paper. This value is approximately half of the $thresh_{utter}$.

If the confidence of the output of the segmentalized DSSR is more than the $thresh_{comb}$, we use the DSSR with all command vocabularies. The reason is that segmentalized DSSRs select an incorrect output occasionally because they consist of many DSSRs. Therefore, we select the output as the final output in this method in the case that the output from the segmentalized DSSR is identical with that of the DSSR with all commands. By combining the two types of DSSRs, we receive benefits of the high word recognition accuracy of the segmentalized DSSRs and the high selection accuracy with the DSSR with all commands.

Table 1. The utterances

Commands	Chats
Big up	Hello
Rotating the images	Thank you
50%	Ok I'll be there now

Our method was based on an assumption that there is a relation in the input sequence from users. For example, users frequently input the parameter value, such as "50%", after speaking command names, such as "Scaling down", to a system. However, this assumption is not always correct. Occasionally, users might input the parameter values before speaking command names: e.g., "200%. Big up the image." In another situation, users input the command name and the parameter value at the same time: e.g. "Rotate 45 degrees clockwise." We deal with these problems. For opposite input sequences, we applied the following rules to our method.

1. Our method uses all DSSRs in the first layer
2. If the output of the first layer is words for parameter values and the confidence is high ($ED_{utter} < 0.14$), our method waits for the next utterance. Otherwise reject the output.
3. If the next utterance is related with the output generated from the previous utterance, our method accepts the two inputs.

For the mixed order utterances, we added recognizers that accept the mixed grammar patterns, such as "COMMAND with PARAMETER", in the first layer. For the added recognizers, we applied the constraint of the confidence to the output selection process. In other words, our method accepts the output from the recognizers for the mixed order utterances only if the confidence is high.

4 Experiments

In this section, we evaluated our methods with 88 utterances about commands and 60 out-of-domain utterances such as greetings. Table 1 shows examples of commands and out-of-domain utterances, namely chats, in the experiment. The number of trials was 5 times and the number of test subjects was 6. For the opposite patterns and mixed order utterances, we used 20 utterances such as "Rotate 90 degrees clockwise." For the additional test data, the number of trails was 5 and the test subject was 1 person.

We used Julius as the LVCSR and Julian as the DSSR [7]. In this experiment, the proposed method contained three layers, one command layer with 10 DSSRs and two parameter layers with 16 DSSRs. We evaluated two criteria as follows:

$$Accuracy = \frac{\# \text{ of commands recognized correctly}}{\# \text{ of commands}}$$

$$ChatDetect = \frac{\# \text{ of chats detected correctly}}{\# \text{ of chats}}$$

Table 2. The experimental result

Method	Accuracy	ChatDetect	Opposite	MixUtter
Non-Hierarchical	85.0	83.6	-	-
Hierarchical	89.0	91.1	-	-
Hierarchical+	88.6	90.3	70.7	74.0

where the “commands recognized correctly” denotes that the method detected a command utterance as “command” correctly and recognized the command correctly. The “chats detected correctly” denotes that the method detected a chat utterance as “chat” correctly. We ignored the word recognition accuracy for the chat utterances because the word accuracy for the chats was out-of-topic in this paper.

Table 2 shows the experimental result. In the table, “Non-Hierarchical” denotes a non-hierarchical method based on the OGSS model. In other words, it consisted of two speech recognizers: one LVCSR and one DSSR with all command and parameter vocabularies for the application. “Opposite” and “MixUtter” are the accuracy of the opposite input sequences and the accuracy of the mixed order utterances. “Hierarchical+” denotes a method with rules for opposite patterns and mixed order utterances. Note that the Non-Hierarchical and Hierarchical could not handle the “Opposite” and “CombUtter”. Although the accuracy and ChatDetect rates decreased slightly, the additional rules were effective for the input patterns. The hierarchical methods outperformed the non-hierarchical method. This result shows the effectiveness of the proposed methods.

The key point of our method was the $thresh_{comb}$. We examined the changes of the accuracy and chatDetect rates. Figure 3 shows the experimental result. If the $thresh_{comb}$ was large, the chatDetect rate decreased dramatically. However, the accuracy and chatDetect rates were stable in the case that the $thresh_{comb}$ was small. This result shows the robustness of our method.

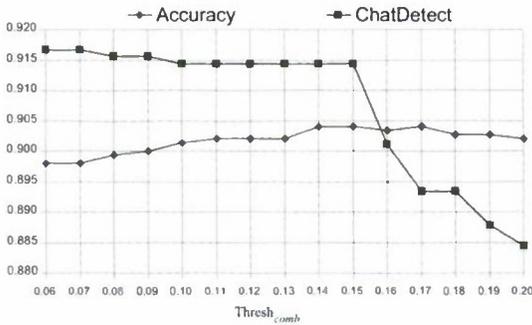


Fig. 3. The threshold

5 Related Work

The system request discrimination method proposed by Sako et al. [4] was based on AdaBoost. The hierarchical method by Lane et al. [6] was based on SVMs. Isobe et al. [8] have proposed a multi-domain speech recognition system based on the model likelihoods of the different domain specific language models. In general, the systems in the previous studies need to recalculate a model to select an output or the current topic. Moreover, machine learning techniques generally need a large amount of training data to generate a classifier with high accuracy. Our method only changes three thresholds.

Komatani et al. [3] have reported an utterance verification method based on difference of acoustic likelihood values computed from two recognizers. Using the difference of acoustic likelihood is adequate for the verification task. Combining the method based on acoustic likelihood with our method is one future work.

6 Conclusions

In this paper, we proposed a hierarchical approach to understand speech inputs. In addition, our method handled opposite input sequences and mixed order utterances. Our method outperformed the non-hierarchical method.

In this paper, we focused on only the selective usage of the multiple speech recognizer. Shimada et al. [9] have reported an integrative usage (an anaphora resolution task) of the OGSS model. The context information recognized by the LVCSR in chats is often important for more deep speech understanding. Future work includes acquisition of the context information from input sequences and the effective utilization of it.

References

1. Bouwman, C., Sturm, J., Boves, L.: Incorporating confidence measures in the dutch train timetable information system developed in the arice project. In: Proceedings of ICASSP (1999)
2. Komatani, K., Kawahara, T.: Flexible mixed-initiative dialogue management using concept-level confidence measures of speech recognizer output. In: Proceedings of COLING 2000, vol. 1, pp. 467–473 (2000)
3. Komatani, K., Fukubayashi, Y., Ogata, T., Okuno, H.G.: Introducing utterance verification in spoken dialogue system to improve dynamic help generation for novice users. In: Proceedings of the 8th SIGdial Workshop on Discourse and Dialogue, pp. 202–205 (2007)
4. Sako, A., Takiguchi, T., Ariki, Y.: System request discrimination based on adaboost. In: IPSJ Technical Report. SIG-SLP64, pp. 19–24 (2006)
5. Shimada, K., Horiguchi, S., Endo, T.: An effective speech understanding method with a multiple speech recognizer based on output selection using edit distance. In: Proceedings of the 22nd Pacific Asia Conference on Language, Information and Computation (PACLIC22), pp. 350–357 (2008)

6. Lane, I.R., Kawahara, T., Matsui, T., Nakamura, S.: Dialogue speech recognition by combining hierarchical topic classification and language model switching. *IEICE Transaction on Information and Systems*, ED 88(3), 446–454 (2005)
7. Lee, A., Kawahara, T., Shikano, K.: Julius - an open source real-time large vocabulary recognition engine. In: *Proceedings of Eurospeech*, pp. 1691–1694 (2001)
8. Isobe, T., Itou, K., Takeda, K.: A likelihood normalization method for the domain selection in the multi-decoder speech recognition system. *IEICE Transaction on Information and Systems (Japanese Edition)* 90(7), 1773–1780 (2007)
9. Shimada, K., Uzunaki, A., Kitajima, M., Endo, T.: Speech understanding in a multiple recognizer with an anaphora resolution process. In: *Proceedings of the 11th Conference of the Pacific Association for Computational Linguistics (PACLING 2009)*, pp. 262–267 (2009)

Author Index

- Abe, Shigeo 487
Al-Obaidi, Ahmed A. 421
- Ban, Sang-Woo 547
Barczak, Andre L. 498
Baxter, Rohan 372
Benferhat, Salem 14
Bollegala, Danushka 595
Bülthoff, Heinrich H. 1
- Cai, Xiaoyan 27
Cao, Longbing 315
Chan, Chee Seng 421, 498
Chen, Shi 256
Chen, Songcan 304
Choe, Yoonsuck 397
Choi, Heeyoul 397
Choi, Seungjin 397
Cho, Sung-Bae 467, 643
Chuang, Lewis L. 1
- De Bruyne, Steven 583
Desrosiers, Christian 39
Dillon, Tharam S. 194
Ding, Yulin 589
Duc, Nguyen Tuan 595
- Ekbal, Asif 52
Endo, Tsutomu 706
- Fukui, Ken-ichi 649
- Garbe, Christoph S. 52
Gelain, Mirco 64
Gerdes, Martin 327
Goto, Tomokazu 595
Granmo, Ole-Christoffer 327
Gretton, Charles 231
Grčar, Miha 219
Guan, Cuntai 535
Gu, Ming 182
Guo, Songshan 256
Gu, Tianlong 384
- Hadzic, Fedja 194
Ha, Jung-Woo 76
Hanapiah, Fazli 421
Handa, Hisashi 433
Han, Yong-Jin 409
Hemer, David 589
Heo, Min-Oh 88
Hong, Gunwon 123
Ho, Weng Luen 601
Hsu, Wen-Chuin 699
Huang, B.Q. 668
Huang, Chung-Hsien 699
Huang, Heqing 359
Huang, Joshua 292
Huang, Wei 608
Huang, Xiaodi 268
Huang, Y. 668
Hwang, Young-Sook 123
- Ishizuka, Mitsuru 4, 595
- Jang, Young-Min 445
Jiang, Lijun 687
Jiang, Yuan 280
John Oommen, B. 327
- Kang, Myunggu 88
Kang, Yoonseop 397
Karypis, George 39
Katake, Anup 397
Kechadi, M.-T. 668
Kim, Byoung-Hee 76
Kim, Donghyun 456
Kim, Kangil 100
Kim, Kee-Eung 614
Kim, Kweon-Yang 409, 637
Kim, Kye-Sung 112
Kim, Min-Jeong 123
Kim, Minook 547
Kim, Sang-Bum 123
Kimura, Masahiro 244
Kim, Youngwook 614
Kubota, Naoyuki 558
Kurihara, Satoshi 649

- Lavrač, Nada 219
 Le, Bac Hoai 477
 Lee, Bado 76
 Lee, Beom-Hee 558
 Lee, Hyoung-Gyu 123
 Lee, Jae-Kul 637
 Lee, Jiann-Der 699
 Lee, Minhó 445, 547
 Lee, Sang-Jo 112, 409, 637
 Lee, Seung-Hyun 467
 Lee, Wono 547
 Le, Thai Hoang 477
 Leung, Ho-fung 510
 Leung, Maylor Karhang 146
 Liao, Zhihua 620
 Li, Chungping 182
 Li, Li 134
 Lim, Andrew 256
 Lim, Sungsoo 467
 Lin, Fen 359
 Lin, Min 327
 Lin, Weiqiang 372
 Liu, Jigang 146
 Liu, Li 315
 Lin, Wei 157, 170
 Liu, Yang 620
 Li, Wenjie 27
 Li, Zhoujun 170
- McKay, Bob (R.I.) 100
 Miao, Yajie 182
 Mohd Shaharanec, Izwan Nizal 194
 Moriyama, Koichi 649
 Motoda, Hiroshi 244
- Ng, Michael 292
 Ngiyen, Doan 625
 Nguyen, Kiem-Hieu 631
 Nguyen, Thanh Tran 477
 Noh, Tae-Gil 637
 Nunao, Masayuki 649
- Ock, Cheol-Young 631
 Ogino, Hiroki 206
 Ohara, Kouzon 244
 Oh, Keunhyun 643
 Ohwada, Hayato 351
 Okumura, Manabu 681
 Orgun, Mehmet A. 372
- Ortiz B., Simón E. 649
 Ozawa, Seiichi 445, 487
- Park, Hyeyoung 456
 Park, Hyung-Min 547
 Park, Seong-Bae 112, 409, 637
 Park, Se Young 112, 409, 637
 Perrussel, Laurent 608
 Pham, Duc Nghia 231
 Pini, Maria Silvia 64
 Plastria, Frank 583
 Podpečan, Vid 219
 Punithan, Dharani 100
- Qian, Junyan 384
 Qiao, Lishan 304
 Quek, Chai 523, 535, 601
- Reichert, Frank 327
 Reyes, Napoleon H. 498
 Richards, Debbie 655
 Rim, Hae-Chang 123
 Robinson, Nathan 231
 Rossi, Francesca 64
- Sadat, Fatiha 662
 Saha, Sriparna 52
 Saito, Kazumi 244
 Sato, T. 668
 Sattar, Abdul 231
 Schuster, Mike 8
 Shi, Daming 146
 Shimada, Kazutaka 706
 Shin, Heesang 498
 Shi, Zhongzhi 359
 Song, Hyun-Je 112
 Song, Xing 693
 Su, Hanjing 292
 Sui, Xin 510
 Su, Jie 693
 Sun, Le 674
- Tabia, Karim 14
 Takamura, Hiroya 681
 Takeuchi, Yohei 487
 Tan, Javan 523
 Taylor, Kerry 134
 Taylor, Meredith 655
 Tung, San Wai 535
 Tung, Whye Loon 601

- Venable, Kristen Brent 64
- Wai, Yan-Yau 699
- Walsh, Toby 11, 64
- Wang, Jim-Jie 699
- Wang, Lei 256
- Wang, Xuesong 384
- Wang, Yang 268
- Wang, Yong 280
- Wanvarie, Dittaya 681
- Wen, Guihua 687
- Wen, Jun 687
- Wen, Si 687
- Wen, Xifeng 327
- Won, Woong Jae 547
- Woo, Jiuseok 558
- Wu, Qingyao 292
- Wu, Yi 280
- Xiang, Shuo 304
- Xiang, Yanping 674
- Xiao, Jianguo 157
- Xiao, Jun 693
- Xu, Xiuhe 693
- Yamauchi, Koichiro 570
- Yang, Jianwu 157
- Yang, Liu 182
- Yang, Shih-Ting 699
- Yang, Yong 315
- Yan, Hualiang 157
- Yazidi, Anis 327
- Ye, Yunming 292
- Yokoyama, Takahiko 706
- Yoshida, Tetsuya 206, 339
- Yoshida, Tsuyoshi 351
- Zeng, Yifeng 674
- Zhang, Byoung-Tak 76, 88
- Zhang, Dapeng 359
- Zhang, Dongmo 608
- Zhang, Du 625
- Zhang, Yan 608
- Zhang, Yihao 372
- Zhang, Zili 620
- Zhao, Lili 182
- Zhao, Lingzhong 384
- Zhou, Zhi-Hua 280
- Zhu, Wenbin 256

Lecture Notes in Artificial Intelligence

Subseries of Lecture Notes in Computer Science

The LNAI series reports state-of-the-art results in artificial intelligence research, development, and education, at a high level and in both printed and electronic form. Enjoying tight cooperation with the R&D community, with numerous individuals, as well as with prestigious organizations and societies, LNAI has grown into the most comprehensive artificial intelligence research forum available.

The scope of LNAI spans the whole range of artificial intelligence and intelligent information processing including interdisciplinary topics in a variety of application fields. The type of material published traditionally includes

- proceedings (published in time for the respective conference)
- post-proceedings (consisting of thoroughly revised final full papers)
- research monographs (which may be based on PhD work)

More recently, several color-cover sublines have been added featuring, beyond a collection of papers, various added-value components; these sublines include

- tutorials (textbook-like monographs or collections of lectures given at advanced courses)
- state-of-the-art surveys (offering complete and mediated coverage of a topic)
- hot topics (introducing emergent topics to the broader community)

In parallel to the printed book, each new volume is published electronically in LNCS Online.

Detailed information on LNCS can be found at www.springer.com/lncs

Proposals for publication should be sent to
LNCS Editorial, Tiergartenstr. 17, 69121 Heidelberg, Germany
E-mail: lncs@springer.com

ISSN 0302-9743

ISBN 978-3-642-15245-0



9 783642 152450

**Lecture Notes in
Artificial Intelligence**

Lecture Notes in Computer Science

› springer.com