ICTNET at Web Track 2012 Ad-hoc Task

Heyuan Li[1, 2], Yuanhai Xue[1, 2], Shaohua Guo[1, 2], Feng Guan[1, 2], Xiaoming Yu[1], Yue Liu[1], Xueqi Cheng[1]

1. Institute of Computing Technology, Chinese Academy of Sciences, Beijing, 100190

2. Graduate School of Chinese Academy of Sciences, Beijing, 100190

Abstract

In this paper, we report our experiments at Ad-hoc task, Web Track 2012. In this year, we attempt to use new web parser with noise elimination. The Conditional Boolean BM25 was used as major ranking function. We also introduce Learning-To-Rank to combine multiple features together for ranking, but the performance was poor due to the low quality of training data.

1. Introduction

Ad-hoc task investigates the performance of search over a static set of documents using previously-unseen topics. The ClueWeb09 Dataset [1] and its derived data were still used this year. The topic used this year was the same as NTCIR-10, which was shorter and more common than 2011's Ad-hoc task. This paper is organized as follows. In Section 2, we discuss the workflow of building index, including parser, data processing, and building. The retrieval models and Learning-To-Rank are described in Section 3. In Section 4, we report the evaluation results and discuss the performance this year. Finally, we conclude our work in Section 5.

2. Building Index

2.1 New web parser with noise elimination

We use the new version of web parser to analysis the web page and extract the text. Many low-quality web pages in ClueWeb09 were filled with advertising and spam. Last year, we use naïve html parser that treat full page as content, which would bring in spam and noise [2]. The new parser proposed this year could remove the noise and spam parts in web page based on DOM characteristic. For example, many paragraphs (<p>) appear continuous trend to be the pure, non-spam text. On the other hand, advertisement and navigation usually consist of intensive hypertext references (<a> tag). We apply the new parser to extract TREC-ID, URL, title, pure content and anchor text.

2.2 Anchor Text

As we discussed last year, search on reverse anchor text could significant improve the retrieval performance [2]. In year 2011, we use anchor text data by Dang [3] and treat each unique pair of <URL, anchor text> as individual document. This year, we use the same original data but employ a map/reduce workflow to merge the anchor text for the same URL. At the map side, we compute the hash value for URL and collect anchor text with count. For the reduce task, we joint the anchor text to form a bigger text for the same key (same URL) and repeat it for many times if the count value is bigger than 1. The reduced text value was seen as a document field for this URL.

2.3 Index Building

This year, we use GolaxyDT2 [5], a real-time distributed search platform. As the data was already generated, we select the distributed mode instead of real-time mode build to speed up the procedure. Firstly, we combine all the fields to form a structured XML documents. The document was consist of 6 fields, including TREC-ID, URL, title, pure content, reverse anchor text and spam value [4]. Secondly, we setup a distributed file system on a 10-servers cluster and copy the structured data to DFS. Then, we deploy the GolaxyDT2 across the cluster. The master controller distributes the data list to each GolaxyDT2 index builder and start the building remotely. The whole building process takes 10 hours to finish.

Report Documentation Page

Form Approved
OMB No. 0704-0188

Public reporting burden for the 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 this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 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 a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE **NOV 2012**	2. REPORT TYPE	3. DATES COVERED **00-00-2012 to 00-00-2012**

4. TITLE AND SUBTITLE **ICTNET at Web Track 2012 Ad-hoc Task**	5a. CONTRACT NUMBER
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Chinese Academy of Sciences,Institute of Computing Technology,Beijing, 100190,**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR’S ACRONYM(S)
	11. SPONSOR/MONITOR’S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

Presented at the Twenty-First Text REtrieval Conference (TREC 2012) held in Gaithersburg, Maryland, November 6-9, 2012. The conference was co-sponsored by the National Institute of Standards and Technology (NIST) the Defense Advanced Research Projects Agency (DARPA) and the Advanced Research and Development Activity (ARDA). U.S. Government or Federal Rights License

14. ABSTRACT

In this paper, we report our experiments at Ad-hoc task, Web Track 2012. In this year, we attempt to use new web parser with noise elimination. The Conditional Boolean BM25 was used as major ranking function. We also introduce Learning-To-Rank to combine multiple features together for ranking, but the performance was poor due to the low quality of training data.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**3**	

3. Retrieval Models and LTR

3.1 Conditional Boolean BM25 Model

Okapi BM25 is a widely used probabilistic retrieval model [6], which is designed for non-structure text. In order to apply this model to structured document with many fields, we combine Boolean model and BM25 Model this year. Firstly, we search on each field using Boolean operator and BM25 separately. For the short fields, such as title and reverse anchor text, we apply OR operator before BM25 ranking. After that, documents that any query words occurs at least once is selected and ranked. For the long fields, such as content, we perform the similar workflow, but use AND operator instead of OR. Secondly, we combine different fields' document together by using AND operator. Documents that exist in all fields rank list would remain to form the final list. Also, we add all part's BM25 score together to obtain the final score and rank by score.

The short field, for instance, AND operator would lead to better accuracy. But for some topic, apply AND operator may cause very few matched documents. Therefore, we propose a modified model this year. For each topic, we use AND operator on title field. If the matching documents is less than threshold, we switch to OR operator. For content field, the operator is always AND. We call this Conditional Boolean BM25 Model and use it as baseline this year.

3.2 Learning to rank

Learning to rank (LTR) introduces machine learning to retrieval ranking problem. It uses supervised or semi-supervised learning to automatically construct a ranking model from training data. LTR is an effective way to combine multiple ranking features together. We choose 11 features and classify them in three types. The Probabilistic Retrieval Model, which is the BM25 score for title, URL, content, reverse anchor fields. The Language Model, including query like-hood and KL-divergence model for title, URL, content and reverse anchor text fields. The content quality feature, such as the spam score and PageRank for document.

Because the lack of feature values that we proposed, we can't use public available training data such as LETOR [8]. Our training data is generated base on the relevance judgment for TREC 2009 and TREC 2010. As a matter of fact, this is a low quality training data, for there are only 98 topics in the dataset and very few relevance documents for each topic. We use RankBoost [9] algorithm this year. For training, we use 5-folds cross validation and the output model is used for ranking directly.

3.3 Boost Wikipedia Result

In many commercial search engines, such as Google, the high quality document is boosted to the first place. This year, we explore the feasibility of boost Wikipedia result. Two indexes are built separately. The first one is INDEX_WIKI, built for pages in ClueWeb09 under the en.wikipedia.org domain. The other pages in ClueWeb09 are built in index INDEX_NORMAL. For INDEX_WIKI, we use Boolean Retrieval Model in URL field. For INDEX_NORMAL, the model described in Section 3.2 is used to rank the result. The final submission is a combination of the two indexes above. For each query, we search across INDEX_WIKI and choose the top one document if any matches. Then search across INDEX_NORMAL and fill the other positions.

4. Results and Discussion

In the first run, ICTNET12ADR1, we use Conditional Boolean BM25 Model described in Section 3.1. Then the matching Wikipedia result is boosted to the first position as shown in Section 3.3. This run is the baseline and we use it to validate the effectiveness of other runs.

In the run ICTNET12ADR2, the workflow was nearly the same as ICTNET12ADR1. But we use the new web page parser described in Section 2.1. We submit this run to investigate the influence of noise elimination for Ad-hoc task.

The last run, ICTNET12ADR3, was generated by Learning-to-rank, as described in Section 3.2. Many features are combined together and a ranking model is automatic generated to form the final result. This is our first attempts that introduce Learning-to-rank for Ad-hoc task recent years.

Run	Rel. Docs.	ERR@20	nDCG@20	P@10	P@20	MAP
ICTNET12ADR1	1398	0.2075	0.1162	0.2860	0.2610	0.0908
ICTNET12ADR2	1337	0.2149	0.1101	0.2680	0.2570	0.0783
ICTNET12ADR3	1210	0.1983	0.0934	0.2280	0.2060	0.0612

Table 1: Result of ad-hoc task, TREC 2012

Table 1 summarizes the performance of our ad-hoc submission this year. As shown in the chart, the first run is best on NDCG, P@5, P@20 and MAP, which is against our prediction. The second run with new parser performance best on ERR@20 but not good at other metrics, which means the noise elimination may optimize top result quality but reduce the recall rate. The third run, which uses Learning-to-rank, performs worst this year. As we point out in Section 3.2, the training data used was generated from relevance judgments in year 2009 and 2010, which is of low quality. We'll try to use other training data and continue to explore effective approach to use learning to rank in Ad-hoc task next year.

5. Conclusion

In this paper, we described our experiment in Ad-hoc task. This year, we use new parser to eliminate the noise in web pages. It's good at optimize top result documents, but may lower the recall. Also, we introduce Conditional Boolean BM25 model to rank the result. To improve the quality of top ranks, we boost the matching Wikipedia page, which play a positive role in the submission compared with last year. Last, we use Learning-to-rank, combining multiple features together to form a rank model, but it performs badly due to the low quality of training data.

6. Acknowledgements

We would like to thank all organizers and assessors of TREC and NIST. This work is sponsored by NSF of China Grants No. 60933005, No. 61100083 and No.61173064, and by 242 Program of China Grants No.2011F65.

References

[1] Carnegie Mellon University. The ClueWeb09 Dataset [EB.OL]. http://boston.lti.cs.cmu.edu/clueweb09. 2009.

[2] Heyuan Li, Yuanhai Xue, Xu Chen, etc. ICTNET at Web Track 2011 Ad-hoc Task. Proceedings of the Twentieth Text REtrieval Conference (TREC 2010) [C], Gaithersburg, Maryland, 2010.

[3] Dang, Croft. Anchor Text Query Log for ClueWeb09 [EB/OL]. http://lemurproject.org/clueweb09/anchortext-querylog/. 2010.

[4] Cormack. Waterloo Spam Rankings for the ClueWeb09 Dataset.Waterloo Spam Rankings [EB/OL]. http://plg.uwaterloo.ca/~gvcormac/clueweb09spam/. 2010.

[5] Golaxy. Golaxy DTSearch 2 [EB/OL]. http://www.golaxy.cn/. 2012.

[6] S E Robertson, S Walkery. OKAPI at TREC8[C]. Proceedings of the Eighth Text REtrieval Conference (TREC 1999). Gaithersburg, MD, USA. 1999.

[7] Wikipedia. Learning to rank. http://en.wikipedia.org/wiki/Learning_to_rank [EB/OL]. 2012.

[8] Tao Qin, Tie-Yan Liu. LETOR: Learning to Rank for Information Retrieval [EB/OL]. http://research.microsoft.com/en-us/um/beijing/projects/letor/. 2012.

[9] Yoav Freund, Raj Iyer, Robert E. Schapire, Yoram Singer. An efficient boosting algorithm for combining preferences [J]. The Journal of Machine Learning Research. 2003.