REPORT DOCUMENTATION PAGE

Form Approved OMB NO. 0704-0188

1. REPORT DATE (DD-MM-YYYY)	2. REPORT TYPE	3. DATES COVERED (From - To)
17-11-2015	Ph.D. Dissertation	-

4. TITLE AND SUBTITLE
Manging a Uer's Vulneratility on a Social Networking Site

5a. CONTRACT NUMBER
W911NF-11-1-0517

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
611102

6. AUTHORS
Pritam Gundecha

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES AND ADDRESSES
Arizona State University
ORSPA
AZ Board of Regents on behalf of Arizona State Unive
Tempe, AZ 85287 -6011

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS (ES)
U.S. Army Research Office
P.O. Box 12211
Research Triangle Park, NC 27709-2211

10. SPONSOR/MONITOR'S ACRONYM(S)
ARO

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
60361-LS.32

12. DISTRIBUTION AVAILIBILITY STATEMENT
Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES
The views, opinions and/or findings contained in this report are those of the author(s) and should not contrued as an official Department of the Army position, policy or decision, unless so designated by other documentation.

14. ABSTRACT
Users often join an online social networking (OSN) site, like Facebook, to remain social, by either staying connected with friends or expanding social networks. On an OSN site, users generally share variety of personal information which is often expected to be visible to their friends, but sometimes vulnerable to unwarranted access from others. The recent study suggests that many personal attributes, including religious and political affiliations, sexual orientation, relationship status, age, and gender, are predictable using users' personal data from an OSN site. The majority of users want to remain socially active, and protect their personal data at the same time. This tension

15. SUBJECT TERMS
Security, User Privacy, User Vulnerability

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	15. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Huan Liu
UU	UU	UU	UU		19b. TELEPHONE NUMBER 480-727-7349

Report Title

Manging a Uer’s Vulneratility on a Social Networking Site

ABSTRACT

Users often join an online social networking (OSN) site, like Facebook, to remain social, by either staying connected with friends or expanding social networks. On an OSN site, users generally share variety of personal information which is often expected to be visible to their friends, but sometimes vulnerable to unwarranted access from others. The recent study suggests that many personal attributes, including religious and political affiliations, sexual orientation, relationship status, age, and gender, are predictable using users' personal data from an OSN site. The majority of users want to remain socially active, and protect their personal data at the same time. This tension leads to a user's vulnerability, allowing privacy attacks which can cause physical and emotional distress to a user, sometimes with dire consequences. For example, stalkers can make use of personal information available on an OSN site to their personal gain. This dissertation aims to systematically study a user vulnerability against such privacy attacks.

A user vulnerability can be managed in three steps: (1) identifying, (2) measuring and (3) reducing a user vulnerability. Researchers have long been identifying vulnerabilities arising from user's personal data, including user names, demographic attributes, lists of friends, wall posts and associated interactions, multimedia data such as photos, audios and videos, and tagging of friends. Hence, this research first proposes a way to measure and reduce a user vulnerability to protect such personal data. This dissertation also proposes an algorithm to minimize a user's vulnerability while maximizing their social utility values.

To address these vulnerability concerns, social networking sites like Facebook usually let their users to adjust their profile settings so as to make some of their data invisible. However, users sometimes interact with others using unprotected posts (e.g., posts from a ``Facebook page\footnote{The term "Facebook page`` refers to the page which are commonly dedicated for businesses, brands and organizations to share their stories and connect with people.}"). Such interactions help users to become more social and are publicly accessible to everyone. Thus, visibilities of these interactions are beyond the control of their profile settings. I explore such unprotected interactions so that users' are well aware of these new vulnerabilities and adopt measures to mitigate them further. In particular, {\em are users' personal attributes predictable using only the unprotected interactions}? To answer this question, I address a novel problem of predictability of users' personal attributes with unprotected interactions. The extreme sparsity patterns in users' unprotected interactions pose a serious challenge. Therefore, I approach to mitigating the data sparsity challenge by designing a novel attribute prediction framework using only the unprotected interactions. Experimental results on Facebook dataset demonstrates that the proposed framework can predict users' personal attributes.

Managing a User's Vulnerability on a Social Networking Site

by

Pritam Gundecha

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved March 2015 by the
Graduate Supervisory Committee:

Huan Liu, Chair
Gail-Joon Ahn
Jieping Ye
Geoffrey Barbier

ARIZONA STATE UNIVERSITY

May 2015

ABSTRACT

Users often join an online social networking (OSN) site, like Facebook, to remain social, by either staying connected with friends or expanding social networks. On an OSN site, users generally share variety of personal information which is often expected to be visible to their friends, but sometimes vulnerable to unwarranted access from others. The recent study suggests that many personal attributes, including religious and political affiliations, sexual orientation, relationship status, age, and gender, are predictable using users' personal data from an OSN site. The majority of users want to remain socially active, and protect their personal data at the same time. This tension leads to a user's vulnerability, allowing privacy attacks which can cause physical and emotional distress to a user, sometimes with dire consequences. For example, stalkers can make use of personal information available on an OSN site to their personal gain. This dissertation aims to systematically study a user vulnerability against such privacy attacks.

A user vulnerability can be managed in three steps: (1) identifying, (2) measuring and (3) reducing a user vulnerability. Researchers have long been identifying vulnerabilities arising from user's personal data, including user names, demographic attributes, lists of friends, wall posts and associated interactions, multimedia data such as photos, audios and videos, and tagging of friends. Hence, this research first proposes a way to measure and reduce a user vulnerability to protect such personal data. This dissertation also proposes an algorithm to minimize a user's vulnerability while maximizing their social utility values.

To address these vulnerability concerns, social networking sites like Facebook usually let their users to adjust their profile settings so as to make some of their data invisible. However, users sometimes interact with others using unprotected posts (e.g.,

posts from a "Facebook page[1]"). Such interactions help users to become more social and are publicly accessible to everyone. Thus, visibilities of these interactions are beyond the control of their profile settings. I explore such unprotected interactions so that users' are well aware of these new vulnerabilities and adopt measures to mitigate them further. In particular, *are users' personal attributes predictable using only the unprotected interactions*? To answer this question, I address a novel problem of predictability of users' personal attributes with unprotected interactions. The extreme sparsity patterns in users' unprotected interactions pose a serious challenge. Therefore, I approach to mitigating the data sparsity challenge by designing a novel attribute prediction framework using only the unprotected interactions. Experimental results on Facebook dataset demonstrates that the proposed framework can predict users' personal attributes.

[1]The term "Facebook page" refers to the page which are commonly dedicated for businesses, brands and organizations to share their stories and connect with people.

DEDICATION

To my grandfather, parents and wife.

ACKNOWLEDGEMENTS

First and foremost, I would like to express my sincere gratitude to my advisor, Dr. Huan Liu. Dr. Liu helped me to develop a taste in research, increased my appetite for novel problems, and showed a path to create my own identity in the research community. He was a constant source of invaluable advice to navigate through the academic world. Most importantly, I learned from him many philosophical principles and disciplines that are not only beneficial for academic research, but also apply to many situations in life. I feel very lucky to have him as my advisor and I sincerely hope that we will remain both collaborators and friends for many years to come.

I would like to thank Dr. Geoffrey Barbier (at the Air Force Research Lab) for playing an instrumental role in my early years as PhD student. I will always be grateful for his mentoring, unconditional support and many thoughtful conversations. I would like to thank my thesis committee, Dr. Gail-Joon Ahn and Dr. Jieping Ye, for their assistance, feedback, and insightful comments. My research has greatly benefited from various collaborations over the years.

I would like to thank Dr. Wesley Gifford, Dr. Ashish Jagmohan and Dr. Anshul Sheopuri (at the IBM T.J. Watson Research Center) for providing me the summer internship opportunity in their group and mentoring me on diverse exciting research projects.

I would like to thank my colleagues at the Data Mining and Machine Learning (DMML) Lab for their insightful discussions and encouragements. This would never be possible without their "all for one and one for all" attitude. It has been a great pleasure working with them, particularly, Jiliang Tang, Xia Hu, Shamanth Kumar, Zhuo Feng, Xufei Wang, Suhas Ranganath, Fred Morstatter, Huiji Gao, Ali Abbasi, Reza Zafarani, Salem Alelyani, Bill Cole, Ashwin Rajadesingan, Isaac Jones, Suhang Wang, Jundong Li, and Liang Wu. I would like to express my gratitude towards the

Army Research Office (ARO) for their continuous support through the grant (No. 025071) during my entire PhD study.

There are far too many people to whom I owe personal thanks, so I will thank many in bulk - Apurva Sahrabudhe, Arul Jain, Niranjan Kulkarni, and all my unofficial roommates there (including those who also never lived there); my ASU friends, Vinay Hanumaiah, Reiley Jeyapaul and Raju Balakrishnan; my weekend friends, Abhishek, Aboli, Gia, Chrital, Dhara, Diya, Kamalesh, Viral, Devansh, Samrat, Noopor, Sanvi, Neha, Priyanka, and Nilesh, who have made Tempe feel like a home over the past six years. Thank you for everything.

I would like to thank my parents who have been working hard to support me all the way. Their endless love and optimistic attitude toward life taught me how to be enthusiastic and happy no matter what kind of difficulty I encounter in my career. My younger brother, Nikhil, has never left my side and his phenomenal support at the start of my career had an impact like that of the butterfly effect. Also many thanks to my sister-in-law who is always proud of me.

Finally, to my lovely wife Snehal Shingi for her understanding and warm support in the past three years. I owe her too many weekends and holidays.

TABLE OF CONTENTS

Page

LIST OF TABLES viii

LIST OF FIGURES ix

CHAPTER

1 INTRODUCTION 1

2 MANAGING VULNERABILITY OF ACTIVE USERS WITH PUBLIC SETTINGS 5

2.1 Measuring User Vulnerability 8

2.2 Reducing User Vulnerability 14

2.3 Reducing User Vulnerability with Social Utility 17

2.4 Experiments 24

2.4.1 Facebook Dataset 25

2.4.2 Vulnerability Reduction without Social Utility Constraints . . 28

2.4.3 Vulnerability Reduction with Social Utility Constraints 35

2.5 Summary 41

3 IDENTIFYING VULNERABILITY OF ACTIVE USERS WITH PRIVATE SETTINGS 43

3.1 Problem Formulation 44

3.2 Framework for Attribute Prediction: SCOUT 45

3.2.1 Learning a Compact Representation 46

3.2.2 Modeling Correlations 47

3.2.3 The Proposed Algorithm to Learn Compact Representation . 52

3.2.4 Building A Classifier for Attribute Prediction 54

3.3 Experiments 55

3.3.1 Facebook Datasets 55

CHAPTER Page

3.3.2 Performance Evaluation 59
3.3.3 Impact of the Learnt Compact Representation 60
3.3.4 Impact of User-User and Post-Post Correlations 62
3.4 Summary 65
4 LITERATURE REVIEW 66
5 CONCLUSION AND FUTURE WORK 73
5.1 User's Vulnerability Across Platforms 74
5.2 Exploring Inactive Users with Private Settings 75
5.3 Identifying Passive Attackers 76
5.4 Extending User Vulnerability to Cope with Identity Theft 78
5.5 Measuring and Reducing Vulnerability of Active Users with Private Settings 78
REFERENCES 80
BIOGRAPHICAL SKETCH 87

LIST OF TABLES

Table Page

2.1 Unfriending One Vulnerable Friend from (A, B, C) to Minimize User U's Vulnerability while Retaining Socially Valuable Friends. A High P-index Suggests High Vulnerability. Social Utility can be Materialized in Various Forms. In This Work, I Use Three Common Measures: One's Nodal Degree, Tie Strength, and Number of Common Friends. . . 18

2.2 Types of Social Utility Measures 20

2.3 Best Known Approximation Schemes and Bounds for the VMC and VMUC Problems. The Objective Function $\sigma(\cdot)$ is Submodular Provided that the Function $h(\cdot)$ is also Submodular in Terms of Vulnerable Friends. 24

2.4 Statistics for Randomly Selected 300K Facebook Users 27

2.5 The Average V-index Values for the Baseline and Three Different Social Utility Measures. Numbers in Brackets are Percentage Decreases in Each Average V-index Value. The Average V-index Value of All Users Before Unfriending Any Vulnerable Friend is 0.3485. 40

3.1 Statistics of the Facebook Dataset 56

4.1 Summarization of Literature Work on Prediction of Private Attributes. Social Media Related Work is Highlighted in Bold 70

LIST OF FIGURES

Figure		Page
1.1	User Types based on Available Information	3
2.1	Which One Vulnerable Friend to Unfriend from (A, B, C) for User U?	6
2.2	Attributes Statistics of Facebook Dataset.	10
2.3	Relationship Among Index Values for Each User.	12
2.4	Facebook Dataset Fact	26
2.5	Performance Comparisons of V-Indexes for Each User Before (Red) and After (Blue) Unfriending based on Eight Different Algorithms.	29
2.6	Performance Comparisons of Unfriending the Most Vulnerable Friend (Red) with Seven Different Unfriending Ways (Blue).	34
2.7	Impact of New Friendship (Blue) on Users with Low V-Indexes (Red) from Users with High V-Indexes	35
2.8	(The Baseline) Performance Comparison of V-Index Values for Each User Before (Red) and After (Blue) Unfriending the k Most Vulnerable Friends from His Social Network.	36
2.9	Performance Comparisons of V-Index Values for Each User Before (Red) and After (Blue) Unfriending at Most k Vulnerable Friends with the Total Degree of Vulnerable Users as a Social Utility Constraint. I Set an Error Parameter $\epsilon = 0.1$ (Input Parameter to FPTAS) and Retain At Least 90% of the Total Degrees of All the Vulnerable Friends After Unfriending.	37
2.10	Performance Comparison Between the Baseline and the Social Utility Approach.	39
3.1	Prediction Performance of the Proposed Framework	61

Figure Page

3.2 Impact of the Learnt Compact Representation on the Proposed Framework. Note that Sometimes the Classifier Gives the Majority Prediction and I can Not Compute Macro-F1 in This Situation; Hence I Use "N.A." to Denote the Performance in the Table. 63

3.3 Impact of User-User and Post-Post Correlations on Predicting Religious Affiliations, Respectively. 64

5.1 Web Interface of the Provenance Data Collector Tool Showing Attribute Values of President Barack Obama (@barackobama) 76

Chapter 1

INTRODUCTION

Social networking sites, such as Facebook and GooglePlus, have gained popularity in recent years, becoming an integral part of our lives. They enable users to remain social by expanding their ways of communications in sharing news, expressing sentiments, exchanging opinions, and making online friends. However with the presence of adversaries, the convenience of and low barriers of access to social networking sites bring about new challenges.

When a user joins a social networking site to remain social, she also expects the protection of personal information from an unwarranted access. This tension leads to a user's vulnerability, allowing myriads of privacy attacks. Vulnerability can cause physical and emotional distress to users, sometimes with dire consequences. For example, Facebook founder is a victim of stalking and has publicly admitted to emotional distress[1]. In a more serious case of cyberstalking, a perpetrator trolled women's Facebook pages searching for clues that allowed him to take over their email accounts[2]. Furthermore, an unwarranted access to personal data on a social networking site can aid not only the cyberbullying of teenagers but also the cyberstalking and cyberharassment of adults[3].

On a social networking site, an individual user can share a large amount of personal information through channels such as the user's profile, frequent status updates, and

[1] http://www.tmz.com/2011/02/07/mark-zuckerberg-restraining-order-facebook-social-network-santa-clara-county-stalker-letters-priscilla-chan

[2] http://usatoday30.usatoday.com/tech/news/2011-07-23-facebook-stalker-sentenced_n.htm

[3] en.wikipedia.org/wiki/Cyberbullying

posts and subsequent interactions. The owner of the site (Facebook, for example) stores such personal information, whereas some (Facebook) users, including friends, has direct access to it. From a user's vulnerability point of view, this gives rise to two types of concerns. First, sometimes the owner of the site sells the personal data to the third party users, including advertising agencies, for generating more revenue. Second, inadequate privacy mechanisms expose the personal data to an unwarranted access from the malicious users and applications (apps). The focus of this dissertation is exclusively on the second concern, whereas previous research on privacy-preserving data-mining [23] proposes different techniques to address the first concern.

To alleviate the vulnerabilities, users are often left with profile settings to mark their personal data, including demographic profiles, status updates, lists of friends, videos, photos, and interactions on posts, invisible to others. Also, the amount of shared information varies for different users. Active users generally share more information whereas inactive users share less information. Based on profile settings and amount of available information, users can be categorized into four types: (1) active users with public settings, (2) active users with private settings, (3) inactive users with public settings, and (4) inactive users with private settings. Figure 1.1 shows users' classification into four quadrants. Users in quadrant one (Q1 users) generally provides maximum amount of personal information including usernames, demographic attributes, lists of friends, posts and associated interactions, and tagging of friends. In comparison with Q1 users, inactive users in quadrant four (Q4 users) provides less amount of information. Usernames, demographic attributes, and lists of friends are generally available from Q4 users. Users in quadrant two (Q2 users) are trying to be social at the same time marking their profile setting private to secure their personal data. New mechanisms such as Facebook page allow Q2 users to interact through posts without requiring them to be friends, while keeping their personal information,

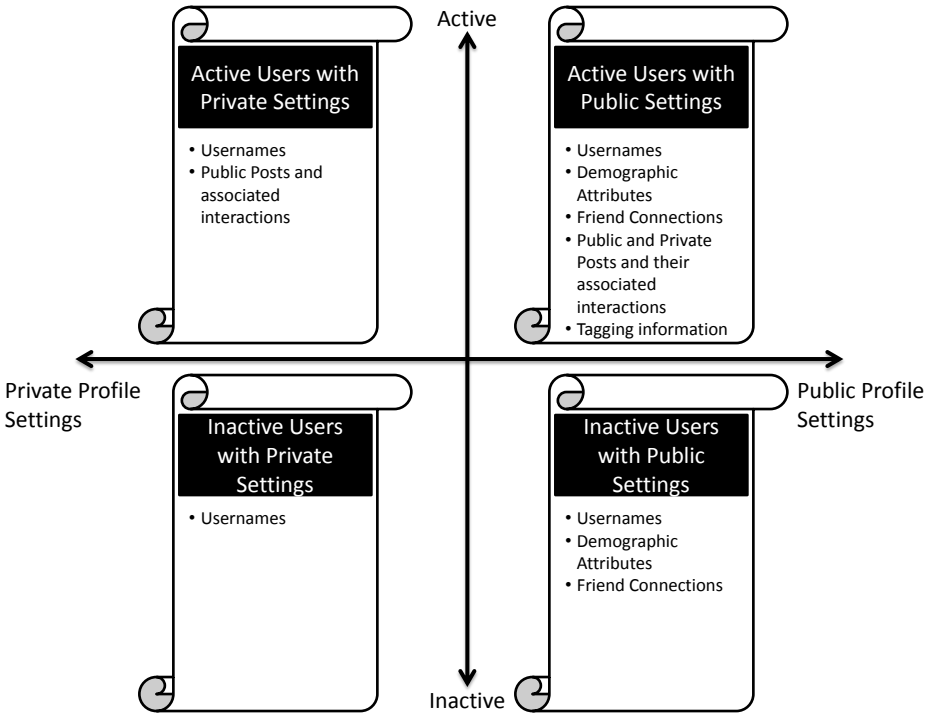


Figure 1.1: User Types based on Available Information

including demographic profiles, lists of friends, and interactions with friends, private. Users' interactions on these pages are often centrally administered and publicly available for everyone. Based on whether a user can control the visibility of her actions, a post can be categorized into two parts: *personal or public post*. Personal post is a post which can be controlled by a user's individual profile settings, otherwise it is referred as a public post. Q2 users often provides usernames, and public posts and associated interactions. The least amount of information is available for users in quadrant three (Q3 users). For a Facebook user, username is the minimum available information.

Based on the literature survey, there are three steps to manage one's vulnerability: (1) identifying, (2) measuring and (3) reducing a user vulnerability. Hence for each of the four types of user, the vulnerability can be managed by addressing these three steps separately. Since active users generously provide information on an OSN site,

this dissertation mainly focuses on active users (i.e., Q1 and Q2 users). Researchers have long been identifying the privacy attacks associated with Q1 users, whereas Q2 users are mostly unexplored. Hence, this dissertation focuses on measuring and reducing Q1 users' vulnerabilities. Furthermore, dissertation also predicts the Q2 users' personal attributes using their interactions on the public posts. To the best of my knowledge, this is the first attempt to identify Q2 users' vulnerabilities.

The remainder of this dissertation is organized as follows. Chapter 2 provides a novel way to measure vulnerabilities of Q1 users, and proposes a mechanism to reduce them further. Chapter 3 identifies the Q2 users' vulnerabilities using interactions on the public posts. Experiments in both Chapters are performed on the large-scale Facebook datasets. A brief literature review is provided in Chapter 4. Finally, conclusion and future work is outlined in Chapter 5.

Chapter 2

MANAGING VULNERABILITY OF ACTIVE USERS WITH PUBLIC SETTINGS

In this chapter, my focus is on active users which mark majority of their profile settings public (i.e., Q1 users). Such users on a social networking site can choose to reveal their personal information using user-names, some demographic attributes, wall-posts and their associated interactions (e.g., likes, comments, reply and shares), lists of friends, and tagging of friends. Researchers have shown that users' personal information could be used in predicting the personal attributes and traits, including gender, age, location, religious and political affiliations, relationship status, sexual orientation, ethnicity, educational level, social ties, parental separation, openness, conscientiousness, extraversion, agreeableness and neuroticism. Chapter 4 reviews the literature on identifying a user's vulnerability arising from the publicly available information. As different users expose to such privacy attacks differently, this Chapter focuses on (1) measuring a user vulnerability, and (2) provide a way to mitigate it. For the rest of this Chapter, active users with public profile settings are simply referred as users, unless otherwise stated.

In this chapter, I show that it is feasible to measure a user's vulnerability based on three factors: (1) user privacy settings can reveal personal information; (2) a user's action on a social networking site can expose their friends' personal information; and (3) friends' action on a social networking site can reveal user's personal information. Based on these three factors, I later show that how user's vulnerability can be measured and assessed. This vulnerability measure enables me to quantify users vulnerability, and identify their vulnerable friends. As we draw parallels between users

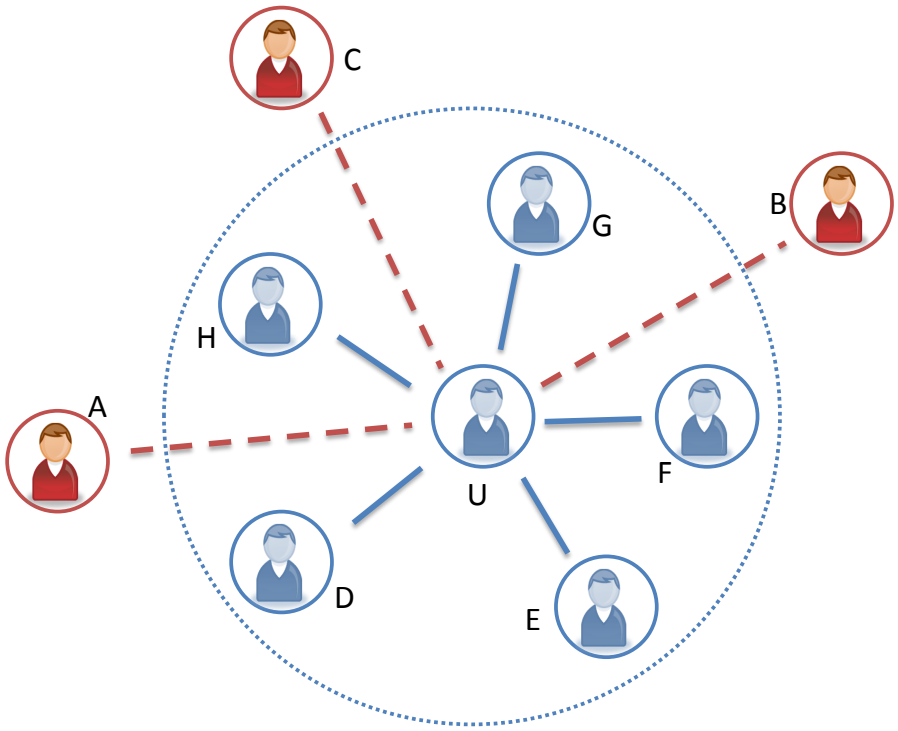


Figure 2.1: Which One Vulnerable Friend to Unfriend from (A, B, C) for User U?

and their friends, I am interested in finding an effective mechanism that could make users less vulnerable. Unfriending with vulnerable friends reduces users vulnerability. This mechanism has been validated using extensive experiments on users and their friends on Facebook.

Sociologists, psychologists and economists [58, 76, 7, 8] have been researching the impact of social interactions on the social utility value of a user and the society. Although unfriending vulnerable friends can reduce vulnerability, this simple strategy can limit social interaction opportunities among users. Besides limiting interaction opportunities, unfriending socially important or valuable friends can backfire and reduce one's social status as well. Social importance can be measured in terms of social utility [48]. One such utility is the nodal degree of a friend. Refer to Figure 2.1: if A is the most vulnerable but also most popular among U's friends, could U unfriend his

other vulnerable friends instead of A in order to reduce vulnerability? Herewith, the additional new challenge is how to maintain low vulnerability and high social utility for a social networking user. The work in this chapter addresses the challenge by developing novel solutions to the following questions without suggesting any structural change to a social networking site.

1. How can we measure and assess user's vulnerability? Is there an effective mechanism to make users less vulnerable?

2. What is the social cost of vulnerability reduction mechanism on user's social utility ? How can we achieve balance between user's vulnerability and social utility?

The rest of the chapter is organized as follows. Section 2.1 studies the collectible statistics from a social networking site and present a quantifiable measure to evaluate users vulnerability and define the problem of identifying vulnerable friends [32]. Section 2.2 proposes a methodology and measures for evaluating whether or not a user is vulnerable and how to adjust a user's network to best deal with threats presented by vulnerable friends [32]. Section 2.3 presents a constrained vulnerability minimization problem. To this end, I formulate two novel optimization problems of vulnerability minimization. I also discuss the hardness of the problem and provide approximation guarantees to efficient algorithms [33]. Section 2.4 conducts an empirical study to evaluate methods that can be manipulated to make users less vulnerable, compare the performance of an optimal algorithm with that of intuitive heuristic methods, and discuss the approach that can be used to assess the impact of new friends to a user's network. I also evaluate methods that make users less vulnerable while retaining acceptable level of social utility values of vulnerable friends. Finally, I summarize the chapter with possible future research directions in Section 2.5.

2.1 Measuring User Vulnerability

Every user on a social networking site can choose to reveal their personal information using a range of attributes. Figure 2.1 shows an illustrative example where a user U has eight friends (A, B, C, D, E, F, G, H). Based on their preferences, friends assumed to be revealing different attributes from the available lists of personal attributes. In this section, I propose a measure to quantify user U's vulnerability. I first divide the user's personal attributes into two sets, *individual and community attributes*. Individual attributes (I-attributes) characterize individual user information, including personal information such as gender, birth date, phone number, home address, group memberships, etc. Community attributes (C-attributes) characterize information about friends of a user, including friends that are traceable from a user's profile (i.e., user's friend list), tagged pictures, wall interactions, etc. These attributes are always accessible to friends but may not to the other users. A user's vulnerability depends on the visibility and exposure of a user's profile through not only attributes settings but also his friends.

Oftentimes users on a social networking site are unaware that they could pose a threat to their friends due to their vulnerability. In this chapter, I show that it is feasible to measure a user's vulnerability based on three factors: (1) user's privacy settings that can reveal personal information; (2) a user's action on a social networking site that can expose their friends' personal information; and (3) friends' action on a social networking site that can reveal user's personal information. Based on these factors, I formally present one of the earliest models for vulnerability reduction.

Definition 1 (I-index) *I-index estimates how much risk to privacy a user can incur by allowing individual attributes to be accessible or visible to other users. A user who ignores or is unaware of privacy settings is a threat to himself. I-index is defined as*

a function of individual attributes (I-attributes). I-index of user u is given by

$$I_u = f(A_u), \tag{2.1}$$

where $I_u \in [0,1]$ *,* $A_u = \{a_{u,i} | \forall i, a_{u,i} \in \{0,1\}\}$ *is I-attribute set for user u, and* $a_{u,i}$ *is a status of a i-th I-attribute for a user u.* $a_{u,i} = 1$ *indicates user u has enabled i-th I-attribute to be visible to everyone otherwise non-visible (may be sensitive for a user). Note a user attribute visible to only friends is marked as disabled.*

Table 2.2 shows statistics of commonly found I-attributes on Facebook (Refer to Section 2.4.1 for details on Facebook dataset consists of 2,056,646 users). The last column in the table lists the percentage of people who enable the particular attribute to be visible. For example, 7,430 (0.36%) Facebook users enabled their mobile phone numbers to be visible. I define the sensitivity (weight), of an attribute as a percentage of non-visibility. Hence, the sensitivity of a mobile phone number according to the proposed Facebook dataset is 99.64. This means that users do not usually disclose their mobile phone number to other users. Users that do disclose phone numbers have a propensity to vulnerability because they disclose more sensitive information in their profiles.

I used normalized weighted average to estimate I-index. I-index for each profile user u is given by,

$$I_u = f(A_u) = \frac{\sum_{i=1}^{n} w_i * a_{u,i}}{\sum_{i=1}^{n} w_i}, \tag{2.2}$$

where w_i is the sensitivity (weight) of an i-th I-attribute, n is the total number of I-attributes available via a social networking site profile and $a_{u,i} = 1$ if i-th I-attribute is visible otherwise the attribute is not visible (i.e., sensitive to user u). $I_u \in [0,1]$. $I_u = 1$ indicates user u has marked all I-attributes to be visible. On the other hand, $I_u = 0$ indicates user u has marked all I-attributes to be non-visible.

Attributes	Users #	Per (%)
I-attributes:		
Current City	620,401	30.17
Hometown	727,674	35.38
Gender	1,681,673	81.77
Birthday	67,834	3.30
Relationship status	539,612	26.24
Siblings	244,658	11.90
Education and work	516,848	25.13
Like and interests	1,369,080	66.57
Email	27,103	1.32
Mobile number	7,430	0.36
Website	128,776	6.26
Home address	7,580	0.37
Political Views	24,438	1.19
Religious Views	33,036	1.61

Attributes	User #	Per (%)
Children	86,609	4.21
Networks	284,482	13.83
Parents	73,887	3.49
Bio	199,070	9.68
Interested in	383,724	18.66
Looking for	449,498	21.86
Music	941,340	45.77
Books	281,346	13.68
Movies	574,243	27.92
Television	684,843	33.30
Activities	385,417	18.74
Interests	308,229	14.99
C-attributes:		
Friends trace	1,481,472	72.03
Total users	**2,056,646**	

Figure 2.2: Attributes Statistics of Facebook Dataset.

From the viewpoint of optimization, it is common to use linear sum as an objective function or constraints to reduce the overall complexity of finding the optimal solution (e.g. Linear Programming). Inspired from this, the chapter proposes a linear sum (weighted average) function of individual attributes and their sensitivity weight (percentage of non-visibility) to compute the I-index of a user (Equation (2.2)). The

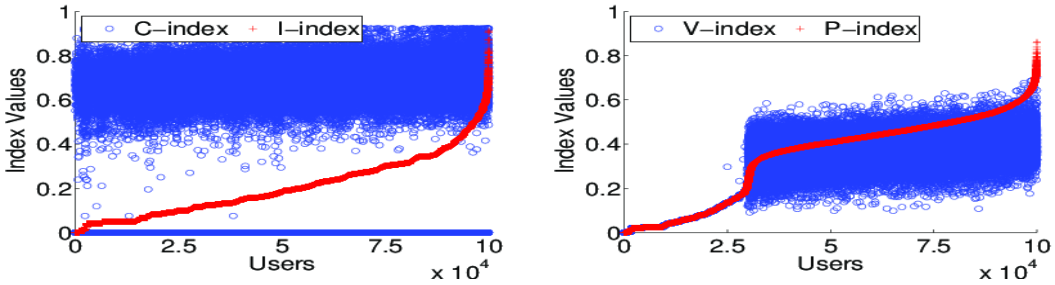
proposed linear sum function is a simple, and captures the intuition that vulnerability of a user increases with more visibility of some attributes. For example, Table 2.2 shows that the profile revealing "religious (1.61%)" and "political (1.19%)" affiliation values should be more vulnerable in comparison with the profile revealing "Gender (81.77%)" and "Relationship status (26.24%)" values.

Definition 2 (C-index) *C-index estimates how much threat a user can pose to their friends by making community attributes accessible or visible to other users. Users who ignore and are unaware of privacy settings of community attributes can create risk to the entire community of friends. C-index is defined as a function of community attributes (C-attributes). C-index for a user u is given by*

$$C_u = g(B_u), \tag{2.3}$$

where $C_u \in [0, 1]$ *,* $B_u = \{b_{u,i} | \forall i, b_{u,i} \in \mathbb{Z}^+\}$ *is C-attributes set for user u,* $b_{u,i}$ *indicates the number of friends affected when a corresponding C-attribute is manifested, and* $\mathbb{Z}^+$ *is the set of positive integers. I ignore attributes marked as non-visible. The Facebook dataset has only one C-attribute (see Table 2.2) which suggests how many friends are traceable (via a friend relationship) from an individual user. 1,481,472 (72.03%) Facebook users in the dataset allowed friends to trace to other users. Thus, a large portion of users are* either not careful or not aware of the privacy concerns of their friends.

A vulnerable user, u, can pose threat to his friends. The amount of the threat increases with the number of friends that are put at risk. However, the rate of the increment decreases as more friends are put at risk. To appropriately represent this threat change, I choose a convex, non-decreasing log function to estimate the threat for each user based on the number friends placed at risk by each C-attribute. Hence,



(a) I-index (Red) and C-index (Blue)

(b) P-index (Red) and V-index (Blue)

Figure 2.3: Relationship Among Index Values for Each User.

C-index for a user u is calculated as

$$C_u = g(B_u) = \frac{\sum_{i=1}^{m} \log(b_{u,i})}{4 * m}, \tag{2.4}$$

where m is the total number of C-attributes possible on a social networking site, and constant 4 is chosen because $C_u \in [0, 1]$ and none of the Facebook users in the dataset has more than 10^4 friends. $C_u > 0$ indicates user u has allowed everyone to trace friends through their own profile. On the other hand, $C_u = 0$ indicates that all the friends (except one) of a user u are non-traceable through a profile.

Fig. 2.3a shows I-index and C-index for randomly chosen 100K Facebook users. Note that users are sorted in ascending order of their I-indexes which gives curve-like impression on plotting I-index. The X-axis and Y-axis indicate users and their corresponding I and C-index values, respectively. Fig. 2.3a demonstrates that for the majority of users, the C-index value is greater than the corresponding I-index value. This highlights the one finding of this chapter that a large portion of users are either not careful or not aware of the privacy concerns of their friends.

Definition 3 (P-index) *It estimates how public (visible) or private (non-visible) a user is on a social networking site. It shows how much an individual user aims to protect himself as well as his friends. P-index is defined as a function of I-index and*

C-index. P-index of user u is given by

$$P_u = j(I_u, C_u), \tag{2.5}$$

where $P_u \in [0, 1]$.

I choose a simple, weighted average function to calculate P-index for each Facebook user in the dataset.

$$P_u = \alpha * I_u + (1 - \alpha) * C_u = \alpha * f(A_u) + (1 - \alpha) * g(B_u), \tag{2.6}$$

where $\alpha \in [0, 1]$. Substituting Eq(2.6) with Eq(2.2) and Eq(2.4), I get

$$P_u = \alpha * \frac{\sum_{i=1}^{n} w_i * a_{u,i}}{\sum_{i=1}^{n} w_i} + (1 - \alpha) * \frac{\sum_{i=1}^{m} \log(b_{u,i})}{4 * m} \tag{2.7}$$

Different users may have different priorities about friends and may have different perspectives about vulnerability. Tunable parameter α can be set to address the needs of different users. For example, one may choose $\alpha < 0.5$ to deemphasize the individual attributes' visibility; or one may choose $\alpha > 0.5$ to emphasize the individual attributes' visibility. For experiments, I set $\alpha = 0.5$ to put equal weights to individual and community attributes.

The P-index address the first two factors of the user vulnerability estimation discussed above, i.e., user's privacy settings (I-index) and actions to expose friends (C-index). Thus, I follow a commonly used optimization formulation as in linear program as I did for I-index. Mathematically, any function which can combine the I-index and C-index and ranges between [0,1] can be used. In this chapter, the above simple weighted average function works well and it can also be easily discerned in applying these indices for various needs.

Definition 4 (V-index) *V-index estimates how vulnerable a user is on a social networking site. Thus far, I have provided three indexes, I-index, C-index, and P-index,*

for a user based on the visibility of I-attributes and C-attributes. Vulnerability of a user depends on privacy settings of self, friends, their friends, and so on. Intuitively, as the distance between a user and other users on a social networking site increases, the marginal risk of vulnerability decreases the further away a user is from a vulnerable user. Hence, I only consider a user and friends in estimating the vulnerability of a user. V-index of a user depends on the P-indexes of friends and him. V-index of user u is defined as

$$V_u = h(P_{F_u \cup \{u\}}), \tag{2.8}$$

where F_u *is the set of friends of user u,* $P_{F_u \cup \{u\}} = \{P_i | i \in F_u \cup \{u\}\}$*, and* $V_u \in [0, 1]$.

I rewrite the above notation without loss of generality as,

$$V_u = h(F_u \cup \{u\}) \tag{2.9}$$

Figure 2.3b shows the P-index and V-index for 100K randomly chosen users (the same users chosen for Figure 2.3a). Note that users are sorted in ascending order of their P-indexes which gives curve-like impression on plotting P-index. The X-axis and Y-axis indicate users and their index values respectively. A simple, weighted average function is used to plot V-index for each user,

$$V_u = \frac{P_u + \sum_{i \in F_u} P_i}{|F_u| + 1} \tag{2.10}$$

The design of V-index function has a direct impact on the complexity of solving the problem of identifying 'k' vulnerable friends (described in detail next). I later show that this problem can be solved in polynomial time, if the V-index is computed as shown in Equation (2.10).

2.2 Reducing User Vulnerability

Next, I will provide the mechanism based on unfriending to reduce user vulnerability.

Definition 5 (A vulnerable friend) *A user's vulnerable friend is defined as a friend whose unfriending will lower the V-index score of a user. The V-index of a user u upon removing the vulnerable friend v is given by*

$$V'_u = h(F_u \cup \{u\} \setminus \{v\}) \tag{2.11}$$

By the definition of a vulnerable friend, $V'_u < V_u$.

The definition of a vulnerable friend can be generalized to k-vulnerable friends.

Definition 6 (k-vulnerable friends) *k-vulnerable friends of a user are k friends whose unfriending will lower the V-index score of a user. The V-index of user, u, upon removing k vulnerable friends $\{v_1, \cdots, v_k\}$ is given by*

$$V'_u = h(F_u \cup \{u\} \setminus \{v_1, \cdots, v_k\}), \tag{2.12}$$

By the definition of k-vulnerable friends, $V'_u < V_u$.

Based on Definitions 5 and 6, a user's friends can be divided into two sets: (1) an initial set of vulnerable friends $D_{u,0}$, who are responsible for increasing the V-index value of user u, and (2) a set of non-vulnerable friends $F_u \setminus D_{u,0}$, who are responsible for decreasing the V-index value of user u. Hence, Eq(2.9) can be rewritten as

$$V_u = h(\{F_u \setminus D_{u,0}\} \cup \{u\} \cup D_{u,0}) \tag{2.13}$$

In order to minimize the user vulnerability[1] function $h(\cdot)$, I have to unfriend vulnerable friends $D_{u,0}$. The user vulnerability minimization problem seeks, for a parameter k from user u, to find a new set of k vulnerable friends $D_u \subseteq D_{u,0}$ to

[1] A user vulnerability can also be minimized by (1) disabling visibility of sensitive user's attributes (such as phone number, email address, home address, etc.) and not exposing friends to others; and by (2) requesting (or negotiating with) the vulnerable friends to lower their vulnerability index.

unfriend. The minimum new V-index for user u is achieved after unfriending the selected vulnerable friends D_u, where $|D_u| \leq k$. The new V-index of user, u, upon removing selected set of vulnerable friends $D_u \subseteq D_{u,0}$, is given by

$$V'_u = h(\{F_u \setminus D_{u,0}\} \cup \{u\} \cup \{D_{u,0} \setminus D_u\}) \tag{2.14}$$

By the definition of k-vulnerable friends, $V'_u < V_u$.

This problem of minimizing the vulnerability of user u is equivalently stated as finding the set of at most k vulnerable friends $D_u \subseteq D_{u,0}$ to unfriend, who are responsible for maximizing the vulnerability of user u.

Let $\sigma(D_u)$ be the estimate how vulnerable user u is due to vulnerable friends D_u. Thus, I maximize function $\sigma : 2^{|D_{u,0}|} \rightarrow R^*$, where $\mathbb{R}^*$ is the set of non-negative real numbers. Note that, $h(D_u)$ is not the same as $\sigma(D_u)$, even though function $\sigma(\cdot)$ depends on function $h(\cdot)$. For example, if $h(\cdot)$ is a simple average function of P-indexes of user and friends, then function $\sigma(\cdot)$ estimates the total P-index value of all the vulnerable friends. In other words, function $\sigma(D_u)$ estimates the vulnerability induced by the vulnerable friends D_u on user u. The Vulnerability Maximization with Cardinality constraint problem (VMC) is formulated as follows

VMC($D_{u,0}$, $\vec{P}$, k, $\mathcal{V}_u$) - *Instance:* The finite set of initial vulnerable friends $D_{u,0} \subseteq F_u$ of user u, P-index $P_i \in [0, 1]$ $\forall i \in D_{u,0}$, a vector $\vec{P} = (P_1, \cdots, P_{|D_{u,0}|})$, and constant $k \in \mathbb{Z}^*$ and $\mathcal{V}_u \in \mathbb{R}^*$. *Question:* Is there a subset $D_u \subseteq D_{u,0}$ such that $|D_u| \leq k$ and $\sigma(D_u) \geq \mathcal{V}_u$?

The above problem can be solved in polynomial time, if function $\sigma(\cdot)$ is a linear function of P-index values of vulnerable friends. The linear function $\sigma(\cdot)$ can be represented as,

$$\sigma(D_{u,0}) = \sum_{i=1}^{|D_{u,0}|} \lambda_i * P_i, \quad \forall i \in D_{u,0}, \lambda_i \in \mathbb{R}^* \tag{2.15}$$

Since vulnerable friends make a user profile less secure, λ_i cannot be a negative real number. For Eq(2.15), the most vulnerable friend $d \in D_{u,0}$ is given by,

$$d = \max_{v \in D_{u,0}} \sigma(D_{u,0}) = \max_i \{\lambda_i * P_i | \forall i \in D_{u,0}\} \tag{2.16}$$

If I repetitively identify the most vulnerable friend d, using Eq(2.16), for k (or n, if $n < k$) times and remove d from $D_{u,0}$ for every run, I can get k-vulnerable friends to maximally reduce user vulnerability.

I do not assume that $\sigma(\cdot)$ is linear. Later, I will discuss how to solve **VMC** problem, when $\sigma(\cdot)$ is non-linear.

So far I aim to reduce the vulnerability of a user without considering its social impact. If the vulnerable user selected to unfriend is also socially valuable, then it could lead to a serious social problem. For example, a user may not want to unfriend his girlfriend, though vulnerable. Next I investigate this problem of user vulnerability reduction with social utility constraints.

2.3 Reducing User Vulnerability with Social Utility

An essential function of social networking sites is to help users to be social. Although unfriending vulnerable friends from a user's social network sometimes can reduce vulnerability, this strategy could sometimes significantly limit social interaction among users.

The social utility can be defined by different measures of social network analysis. In this work, I use three basic measures: nodal degree or simply degree, tie strength, and number of common friends. A friend with high degree means she is a popular one; when two friends have a strong tie [25] or have a large number of common friends, they are very close friends. In other words, I can employ these measures help determine the consequence of unfriending on a user. Unfriending a vulnerable friend with high

Vulnerable friends	P-index	Degree	Tie Strength	#Common Friends
A	0.9	100	0.8	30
B	0.7	200	0.5	50
C	0.5	300	0.3	10

Table 2.1: Unfriending One Vulnerable Friend from (A, B, C) to Minimize User U's Vulnerability while Retaining Socially Valuable Friends. A High P-index Suggests High Vulnerability. Social Utility can be Materialized in Various Forms. In This Work, I Use Three Common Measures: One's Nodal Degree, Tie Strength, and Number of Common Friends.

degree could limit the user's potential to make more friends, which makes user more reclusive and defeats the purpose of social networking. Generally, family members, best friends, and girl or boy friend are examples of strong ties. Though unfriending them could reduce one's vulnerability, it would not be desirable. When the user and his friend share a large number of friends, removing the friend could affect structural balance [20] of the social network and the user's clustering coefficient [78]. Thus, it is essential to consider the social utility of vulnerable friends before unfriending them in order to reduce vulnerability.

There exist many social utility measures that can be categorized into two types [4]: (1) social utility measures related to the vulnerable friends, and (2) social utility measures related to the relationship between user and each of his vulnerable friends. Table 2.2 lists some social utility measures available for a user u to consider while selecting the vulnerable friends $D_u \subseteq D_{u,0}$ to unfriend. Figure 2.1 shows an illustrative example where a user U has three vulnerable friends (A, B, C). Table 2.1 lists users P-indexes (indicating user visibility) and their social utility measures in-

cluding degree, tie strength and number of common friends. If I do not consider the social utility measures of vulnerable friends, the most vulnerable friend A should be removed to minimize user U's vulnerability. If I consider tie strength, user A is the most valuable for user U. Similarly, friends B and C are also valuable because B shares the most number of common friends and C has the highest degree. Under these different circumstances, I now study which friend U should unfriend in order to reduce vulnerability with the constraint of retaining social utility.

Social utility measures related to the initial set of vulnerable friends $D_{u,0}$ of a user u.	Degree (number of friends) of a vulnerable friend. Local clustering coefficient (density) of a vulnerable friend. Number of closed triads of a vulnerable friend. Number of open triads of a vulnerable friend. Number of posts (social activity) by a vulnerable friend. Number of responses (social activity) by a vulnerable friend. Popularity (may be based on replies on posts) of a vulnerable friend. Influence index of a vulnerable friend.
Social utility measures related to the relationship between user u and each vulnerable friend in $D_{u,0}$	Number of common friends between user and vulnerable friend. Tie strength between user and vulnerable friend. Number of uncommon friends between user and vulnerable friend. Number of responses (social activity) by a user u on a vulnerable friend's post. Number of responses (social activity) by a vulnerable friend on a user's post. Homophily (similarity) index between user and a vulnerable friend.

Table 2.2: Types of Social Utility Measures

Next I define the optimization problem for reducing a user vulnerability while retaining socially valuable friends.

Definition 7 (Social Utility Loss) *It estimates how much a user loses on a social networking site after unfriending friends. Social utility loss of user u depends on*

social utility values of u's friends. For a given social utility measure, the social utility loss for a user u after unfriending with a given set of friends $A \subseteq F_u$ is given by

$$L_u = \zeta(S_A), \tag{2.17}$$

where $S_A = \{S_i | i \in A \subseteq F_u, S_i \in \mathbb{R}^*\}$, S_i represents the social utility measure of user i, and $L_u \in \mathbb{R}^*$.

Assuming $\zeta(\cdot)$ to be a simple additive function, I have

$$L_u = \sum_{i \in A \subseteq F_u} S_i \tag{2.18}$$

Similar to the **VMC** problem formulation, the problem of minimizing the vulnerability of user u with social utility constraint can be stated as finding the set of at most k vulnerable friends $D_u \subseteq D_{u,0}$ to unfriend, who are responsible for maximizing the user vulnerability, and minimizing user u's social utility loss. I can formally state the problem of Vulnerability Maximization with minimum social Utility loss and Cardinality constraint (**VMUC**) as

VMUC($D_{u,0}$, $\vec{P}$, $\vec{S}$, k, $\mathcal{V}_u$, $\mathcal{L}_u$): - *Instance:* Given a finite set of initial vulnerable friends $D_{u,0} \subseteq F_u$ of a user u, P-index $P_i \in [0,1]$, $\forall i \in D_{u,0}$, a vector $\vec{P} = (P_1, \cdots, P_{|D_{u,0}|})$, social utility measure $S_i \in R^*$ $\forall i \in D_{u,0}$, a vector $\vec{S} = (S_1, \cdots, S_{|D_{u,0}|})$, and constants $k \in \mathbb{Z}^*$, $\mathcal{V}_u \in \mathbb{R}^*$, and $\mathcal{L}_u \in \mathbb{R}^*$. - *Question:* Find a subset $D_u \subseteq D_{u,0}$ such that $|D_u| \leq k$, $\sigma(D_u) \geq \mathcal{V}_u$, and $\sum_{i \in D_u} S_i \leq \mathcal{L}_u$.

I now focus on solving the **VMUC($D_{u,0}$, $\vec{P}$, $\vec{S}$, k, $\mathcal{V}_u$, $\mathcal{L}_u$)** problem. First, I prove the function $\sigma(\cdot)$ is non-negative, non-decreasing and submodular.

Theorem 1 (Monotonicity) *The function $\sigma : D_{u,0} \rightarrow \mathbb{R}^*$ is monotonically non-decreasing. i.e, $\sigma(A) \leq \sigma(A \cup \{v\})$, where $A \subseteq D_{u,0}$ and $v \in D_{u,0}$.*

Proof. As discussed above, for a user u, each friend can be classified into vulnerable or non-vulnerable friend. From Definition 5, the V-index of a user u decreases upon

removing the vulnerable friend v. Therefore, $h(S \cup A) \leq h(S \cup A \cup \{v\})$, where S and A are sets of non-vulnerable (including user u) and vulnerable friends, respectively. This means that for user u, the vulnerability of set $A \cup \{v\}$ is more than that of set A. Hence, $\sigma(A) \leq \sigma(A \cup \{v\})$. □

Theorem 2 (Submodularity) *If the function $h(\cdot)$ is submodular in terms of vulnerable friends, the function $\sigma(\cdot)$ is submodular, i.e., $\sigma(A \cup \{v\}) - \sigma(A) \geq \sigma(B \cup \{v\}) - \sigma(B)$, where $A \subseteq B \subseteq D_{u,0}$, and $v \in D_{u,0}$.*

Proof. If the function $h(\cdot)$ is submodular in terms of vulnerable friends, the marginal gain in vulnerability of user u, by adding a vulnerable friend v to an initial vulnerable set A, is at least as high as the marginal gain, by adding the same vulnerable node v to an initial vulnerable superset B, i.e., $h(S \cup A \cup \{v\}) - h(S \cup A) \geq h(S \cup B \cup \{v\}) - h(S \cup B)$, where S is the set of non-vulnerable friends (which includes user u), A and B are sets of vulnerable friends, and $A \subseteq B$. This means that the new vulnerable friend v causes more increase when added to a set A than to a superset B. Thus, $\sigma(\cdot)$ is submodular. □

Let us examine the assumption that the function $h(\cdot)$ is submodular in terms of vulnerable friends. Assume that user v is user u's vulnerable friend. If user u has 25 vulnerable friends as opposed to 50, where 25 vulnerable friends are a subset of 50, then based on the assumption about the function $h(\cdot)$, user v is more vulnerable for user u when u has fewer vulnerable friends than when u has more. In other words, the vulnerability of user v can be mitigated due to the presence of more u's vulnerable friends.

Based on Theorems 1 and 2, the **VMC** problem is tantamount to the maximization of non-negative, non-decreasing, submodular function with cardinality constraint. A hill climbing algorithm can solve this problem with provable constant approxima-

tion [57]. First start with an empty output set D_u; add one element from an initial set of vulnerable friends $D_{u,0}$ to the output set that provides the largest marginal increase in the function value; repeat the previous step until all the elements from an initial set of vulnerable friends $D_{u,0}$ are processed or the maximum cardinality bound k is reached. According to [57], this greedy algorithm gives a $(1 - 1/e)-$approximation for maximization of $\sigma(\cdot)$ function with a given cardinality constraint.

Similarly, with Theorems 1 and 2, the **VMUC** problem is equivalent to the maximization of non-negative, non-decreasing, submodular function with knapsack like constraints. The greedy algorithm presented in [70] can be applied to solve the **VMUC** problem with submodular objective function constrained by cardinality and social utility. The proposed algorithm also gives $(1 - 1/e)$-approximation guarantee.

The VMUC problem remains NP-hard [24] even when the objective function $\sigma(\cdot)$ is linear, constrained by social utility and cardinality. It can be reduced to a single dimensional knapsack problem. The following scaling and rounding algorithm is a fully polynomial time approximation scheme [75] (FPTAS) for the VMUC problem with a linear objective function with knapsack like constraints: for each vulnerable user $i \in D_{u,0}$, define new P-index $P'_i = \lfloor \frac{P_i}{K} \rfloor$, where $K = \frac{\epsilon * P}{n}$, $n = |D_{u,0}|$, $P = \sum_{i \in D_{u,0}} P_i$ and a given error parameter $\epsilon > 0$; with the new P-index, using a dynamic programming algorithm similar to the single dimensional knapsack problem, find the most vulnerable set $D_u \subseteq D_{u,0}$ and $|D_u| \leq k$.

For applying the scaling rounding algorithm, P-index values of all vulnerable friends need to be integers. The P-index value of each user $i \in D_{u,0}$ is a non negative real number in the closed range 0 to 1. Since the proposed problem is a discrete optimization problem, I can simply convert these P-index values into integers by shifting the decimal points equally to the right. For experiments, on the **VMUC** problem with linear objective, I multiply each P-index value by 1000 and then take the floor of

the resulting value as new integer value for P-index. Errors caused due to the scaling and rounding are negligible.

Solutions presented for **VMC** and **VMUC** problems, with corresponding assumptions, are summarized in the Table 2.3. The **VMUC** problem for non-linear social utility gain constraints remains an open problem.

		VMC	VMUC Problem
Objective function $\sigma(\cdot)$	Linear	1	FPTAS
	Submodular	$(1 - 1/e)$	$(1 - 1/e)$

Table 2.3: Best Known Approximation Schemes and Bounds for the VMC and VMUC Problems. The Objective Function $\sigma(\cdot)$ is Submodular Provided that the Function $h(\cdot)$ is also Submodular in Terms of Vulnerable Friends.

2.4 Experiments

The proposed methods are demonstrated in practice through experiments using a dataset derived from a real social networking site. The proposed experiments address the challenge of vulnerability reduction with and without social utility constraints. With an approach for identifying vulnerable friends, I set out to investigate the following issues:

- How effective are the measures in reducing vulnerability of users? What is an effective way of reducing one's vulnerability? How does it compare random unfriending in reducing vulnerability of users?

- Do the indexes address the dynamics of social networks? I study the impact of a new friend request and its effect on vulnerability of a user.

- How effective are the unfriending algorithms in recommending at most k (≥ 0) vulnerable friends to minimize user vulnerability while maintaining an acceptable level[2] of social utility loss?

- Does the user vulnerability reduction change significantly for different social utility measures?

- At most, how many vulnerable friends should a user unfriend to achieve a desired vulnerability reduction while maintaining an acceptable level of social utility loss?

Next I discuss the dataset used for experiments, use the proposed index estimation methods in an empirical study in an attempt to address these issues, report preliminary results, and suggest new lines of research in finding vulnerable users.

2.4.1 Facebook Dataset

According to Quantcast[3], over 145 million unique users in United States visit Facebook within a month. This puts Facebook among top 3 websites based on the number of people in the United States who visit each site within a month. Facebook users spend over 700 billion minutes per month. The statistics suggest that Facebook[4] users provide a rich set of personal information through their profile, and social activities. Thus, I use a Facebook dataset for evaluating the proposed work.

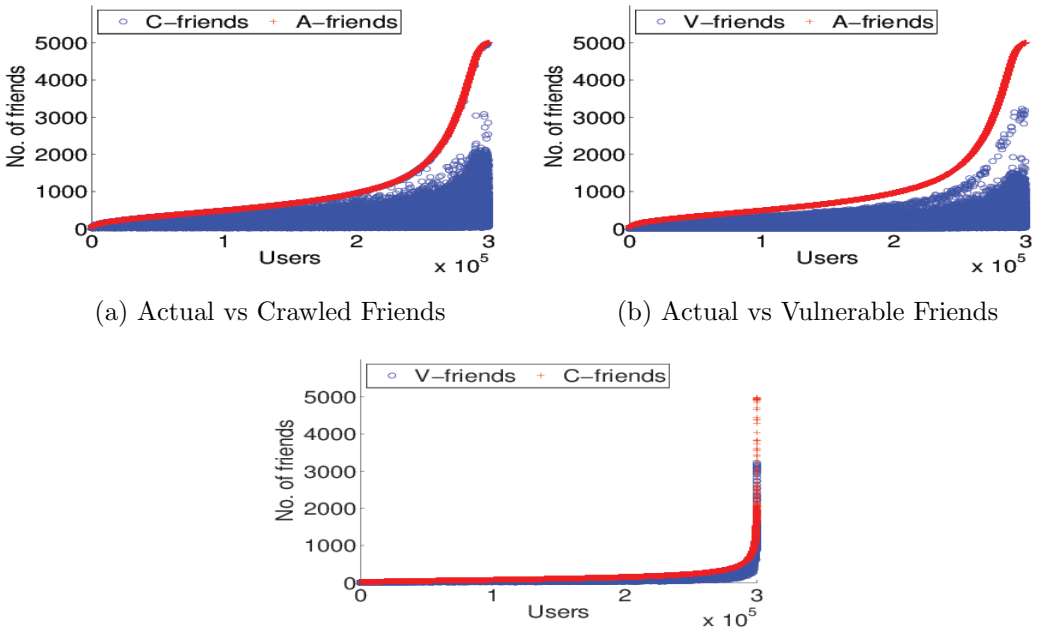
The Facebook dataset[5] is created by crawling Facebook user profiles. Crawling is performed in breadth-first search manner starting from randomly selected users(roots). The dataset contains publicly available profiles as well as network in-

[2]In this work, I set the acceptable level of social utility loss less than or equal to 10%

[3]http://www.quantcast.com/facebook.com, a media sharing and web analytics service company.

[4]http://newsroom.fb.com/content/default.aspx?NewsAreaId=22

[5]I use the same dataset as in [32]



(a) Actual vs Crawled Friends

(b) Actual vs Vulnerable Friends

(c) Crawled vs Vulnerable Friends

Figure 2.4: Facebook Dataset Fact

formation for more than two million users. I design two major tasks: vulnerability reduction without social utility constraints, and vulnerability reduction with social utility constraints. For the first task, I do not filter any Facebook user profiles from the dataset. However, for the second task I remove the Facebook users who do not share their friends' information from the dataset, since unfriending is not possible on such users. Table 2.4 shows the statistics of randomly selected 300K users from the dataset used for the second task.

Facebook dataset	Count
Avg. actual friends per user	1056
Avg. crawled friends per user	154
Avg. vulnerable friends per user	98
Max. actual friends per user	5000
Min. actual friends per user	11
Max. crawled friends per user	4971
Min. crawled friends per user	11
Max. vulnerable friends per user	3222
Min. vulnerable friends per user	0

Table 2.4: Statistics for Randomly Selected 300K Facebook Users

For a given user, I may not obtain the information of all friends due to their privacy settings. Friends for which I obtain the information are referred as crawled friends. Figure 2.4 shows further facts about the randomly selected 300K users. X-axis and Y-axis indicate users and their number of friends, respectively. For simplicity, before plotting Figures 2.4a and 2.4b, I sort all users in the ascending order based on the number of Facebook friends, while for Figure 2.4c, I sort all users in the ascending order based on the number of crawled Facebook friends. Figure 2.4a shows the relationship between actual Facebook friends (red) and crawled friends (blue) in dataset. In experiments, I estimate the V-index of each user as an average of all the P-index values of crawled friends. Based on the V-index, I compute the number of vulnerable friends for each user, as described in Section 2.1. Figure 2.4b shows

the relationship between actual Facebook friends (red) and their vulnerable friends (blue). Figure 2.4c shows the relationship between crawled Facebook friends (red) and their vulnerable friends (blue).

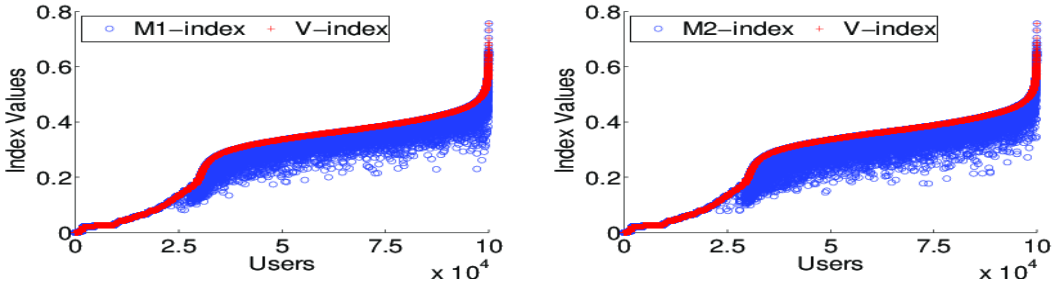
2.4.2 Vulnerability Reduction without Social Utility Constraints

I divide this major task into three experiments to test the (1) impact of different unfriending strategies on users' vulnerability, (2) performance of unfriending the most vulnerable friend with different unfriending strategies, and (3) impact of new friendship on secure users from vulnerable users.

Impact of different unfriending strategies

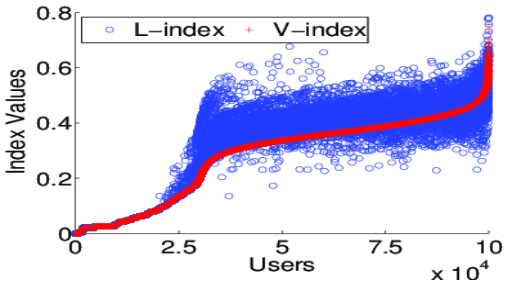
For the first set of experiments, I compare V-index for each of user with two optimal algorithms and six intuitive strategies for unfriending to reduce vulnerability. For each graph in Figure 2.5, the X-axis and Y-axis indicate users and their V-index values, respectively. For simplicity, I sort all users in ascending order based on existing V-index, and then I plot their corresponding V-index before and after unfriending. Figure 2.5 indicates performance of all eight algorithms which will help us to decide whether unfriending makes users more or less vulnerable. The eight algorithms are,

- *Most vulnerable friend.* For a user, the most vulnerable friend is the one whose removal lowers the V-index score the most. For each user, I first find the most vulnerable friend and then estimate the new V-index value (M1-index) after unfriending him/her. As expected, see Figure 2.5a, V-index values for users decrease in comparison with V-index values before unfriending the most vulnerable friend. Unfriending the most vulnerable friend makes all users more secure.

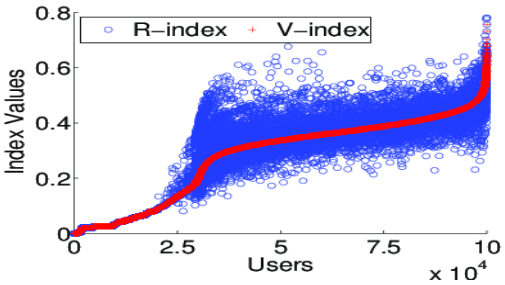


(a) Most Vulnerable Friend

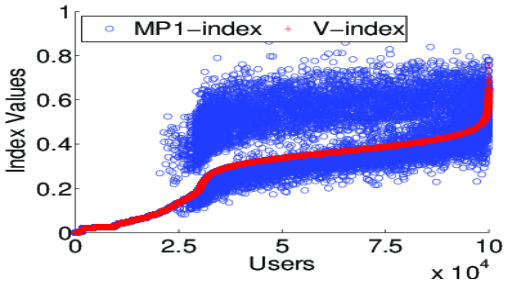
(b) 2 Most Vulnerable Friends



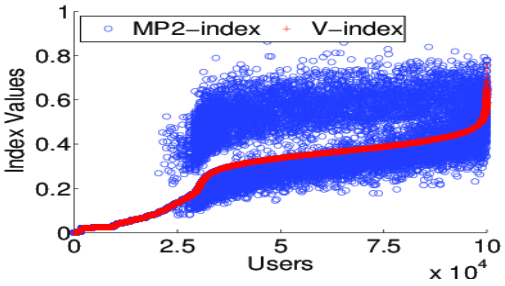
(c) Least V-Friend



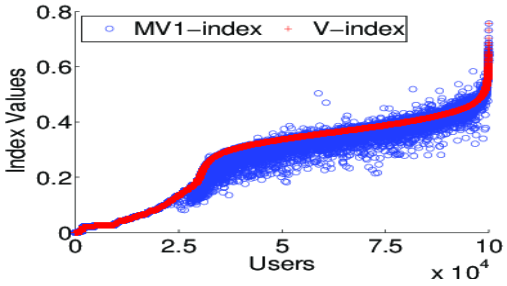
(d) Random Friend



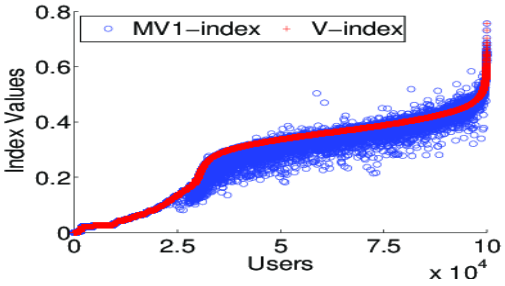
(e) Max P-Friend



(f) 2 Max P-Friends



(g) Max V-Friend



(h) 2 Max V-Friends

Figure 2.5: Performance Comparisons of V-Indexes for Each User Before (Red) and After (Blue) Unfriending based on Eight Different Algorithms.

- *Two most vulnerable friends.* If I sort all of user's vulnerable friends in ascending order based on their new V-indexes (after unfriending), the top two in the list are the two most vulnerable friends. For each user, I first find two most vulnerable friends and then estimate the new V-index value (M2-index) after unfriending them. As expected, see Figure 2.5b, V-index values for all users decrease in comparison with V-index values before unfriending the two most vulnerable friends. Unfriending the two most vulnerable friends also make all users more secure.

- *Least V-friend.* For each user, I choose to unfriend the friend whose V-index is the lowest among all friends. This friend is the least V-friend. V-index values increase for 65% of 100K users, and increase for 43% of the 2M+ users, in comparison with V-index values before unfriending the least V-friend. See Figure 2.5c, L-index refers to the new V-index value after unfriending the least V-friend. $V'_u > V_u$ for some users because, $P_l < V_u$ where P_l is the P-index of the least V-friend. Unfriending the least V-friend does not make all users insecure.

- *Random friend.* For each user, I randomly choose to unfriend a friend. V-index values increase for 24% of 100K users, and increase for 23.5% of the 2M+ users, in comparison with V-index values before unfriending a random friend. See Figure 2.5d, R-index refers to the new V-index value after unfriending a random friend. $V'_u > V_u$ because $P_r < V_u$, where P_r is the P-index of the random friend. Unfriending a random friend does not make all users secure.

- *Max P-friend.* For each user, I choose to unfriend a friend whose P-index is the highest among all friends. V-index values increase for 5% of 100K users, and increase for 11% of the 2M+ users, in comparison with V-index values before

unfriending the max P-friend. See Figure 2.5e, MP1-index refers to the new V-index value after unfriending the max P-friend. $V'_u > V_u$ for some users because $P_{mp_1} < V_u$, where P_{mp_1} is the P-index of the max P-friend. Unfriending the max P-friend makes a majority of users more secure.

- *Two max P-friend.* For each user, I choose to unfriend two friends whose P-index is the highest and second highest among all friends. V-index values increase for 5% of 100K users, and increase for 11% of the 2M+ users, in comparison with V-index values before unfriending the two max P-friends.See Figure 2.5f, MP2-index refers to the new V-index value after unfriending the two max P-friend. $V'_u > V_u$ for some users because $(P_{mp_1} + P_{mp_2})/2 < V_u$, where P_{mp1_1} and P_{mp_2} are P-indexes of the two max P-friends. Unfriending the two max P-friends makes a majority of users more secure.

- *Max V-friend.* For each user, I choose to unfriend a friend whose V-index is the highest among all friends. V-index values increase for 3.6% of 100K users, and increase for 5% of the 2M+ users, in comparison with V-index values before unfriending max V-friend. See Figure 2.5g, MV1-index refers to the new V-index value after unfriending the max V-friend. $V'_u > V_u$ for some users because $P_{mv_1} < V_u$, where P_{mv_1} is the P-index of the max V-friend. Unfriending the max V-friend makes a majority of users more secure.

- *Two max V-friend.* For each user, I choose to unfriend two friends whose V-index is the highest and second highest among all friends. V-index values increase for 2.5% of 100K users, and increase for 5% of the 2M+ users, in comparison with V-index values before unfriending the two max V-friends. See Figure 2.5h, GV2-index refers to the new V-index value after unfriending the two max V-friends. $V'_u > V_u$ for some users because $(P_{mv_1} + P_{mv_2})/2 < V_u$,

where P_{mv_1} and P_{mv_2} are P-indexes of the two max V-friends. Unfriending the two max V-friends make a majority of users more secure.

Performance comparison with the best unfriending strategy

In the second set of experiments, I compare the performance of unfriending most vulnerable friends with the seven intuitive unfriending strategies. For each graph in Figure 2.6, the X-axis and Y-axis indicate users and their associated V-index values after unfriending, respectively. I sort all users in ascending order based on V-index values after unfriending the most vulnerable friend and then plot corresponding V-index based on different unfriending strategies. I find unfriending the most vulnerable friend makes users more secure.

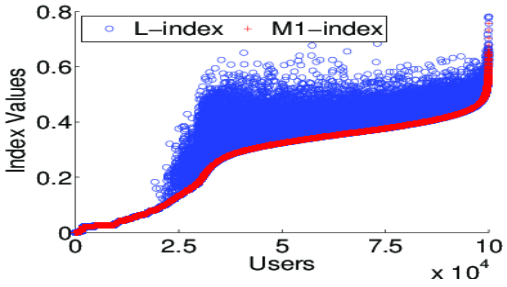
As expected, see Figure 2.6a-2.6d, V-index values for each user based on unfriending the least V-friend, a random friend, the max P-friend, or the max V-friend increase for all users in comparison with their V-index values after unfriending the most vulnerable friend. In the case of unfriending the least V-friend, V-index values increase for 3% of users in comparison with the most vulnerable friend unfriending. Similarly, 1.7% of users increase for a random friend unfriending, 1.7% of users increase for the P-friend unfriend, and 1% of users increase for the V-friend unfriending. Thus, unfriending the most vulnerable friend makes all users more secure than all other schemes.

V-index values for each user based on unfriending the two most vulnerable friends, see Figure 2.6e, do not decrease for 10% of 100K, and 21% of 2M+ users, in comparison with V-index values after unfriending the most vulnerable friend. V-index values for each user based on unfriending the two max P-friend, see Figure 2.6f, do not decrease for 51% of 100K, and 81% of 2M+ users, in comparison with V-index values after unfriending the most vulnerable friend. V-index values for each user based on

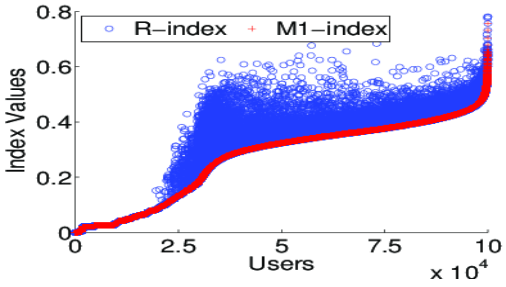
unfriending the two max V-friend, see Figure 2.6g, do not decrease for 90% of 100K, and 75% of 2M+ users, in comparison with V-index values after unfriending the most vulnerable friend.

Impact of new friends

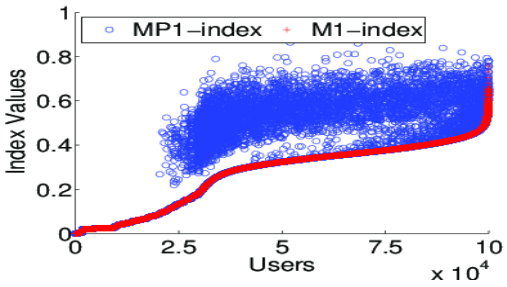
I now investigate the impact of new friendship on two types of secure users from vulnerable users. I select three sets of 10K users from 2M+ Facebook users: (S1) users with high V-indexes, (S2) users with low V-indexes, and (S3) C-attributes enabled users with low V-indexes. I randomly select a vulnerable user (i.e., selected from S1, 10K high V-index users) and a secure user (i.e., selected from S2, 10K low V-index users), and pair them and remove the pair from S1 and S2, respectively, until all 10K users from S1 and S2 are paired. I repeat the same with sets S1 and S3. The two sets of results are shown in Figure 2.7 (a) and (b). For each graph, the X- and Y-axis indicate users and their V-index values before and after the pairing of new friends, respectively. I sort all users in ascending order based on their old V-indexes. As shown in Figure 2.7a, V-indexes of all users of S2 increase significantly and consistently; in Figure 2.7b, V-indexes of users of S3 also increase, but vary from minor to large changes. The larger changes in the latter case occur on those users of S3 with fewer friends. The results in Figure 2.7 confirm that less vulnerable users can become more vulnerable if they are careless when making new friends, and reclusive users are more sensitive to the choice of new friends than less reclusive ones.



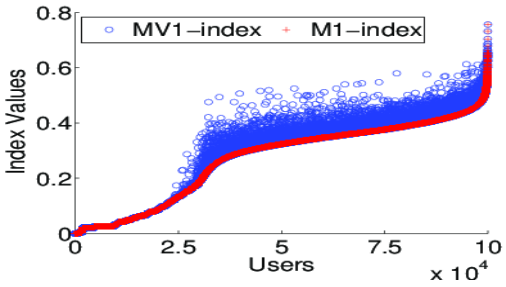
(a) Least V-friend



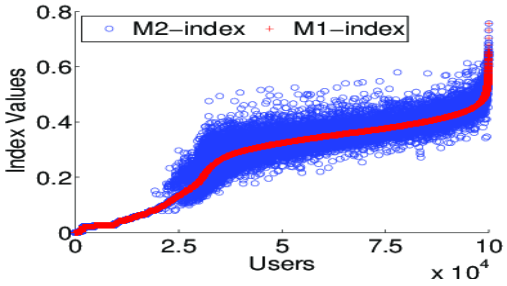
(b) Random Friend



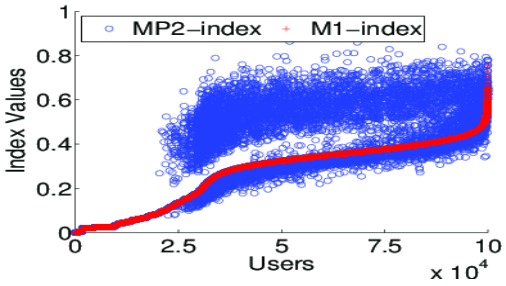
(c) Max P-Friend



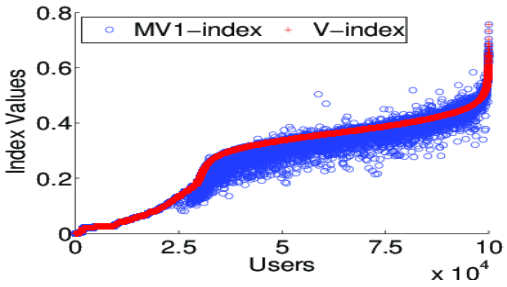
(d) Max V-Friend



(e) 2 Most Vulnerable Friends

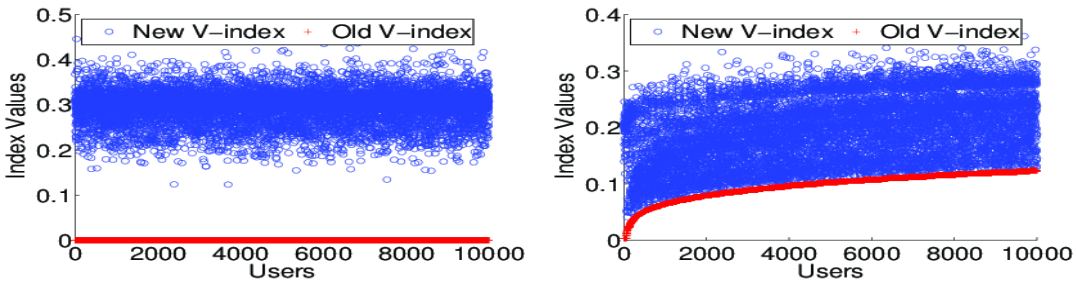


(f) 2 Max P-Friends



(g) 2 Max V-Friends

Figure 2.6: Performance Comparisons of Unfriending the Most Vulnerable Friend (Red) with Seven Different Unfriending Ways (Blue).



(a) C-Attributes are Not Visible

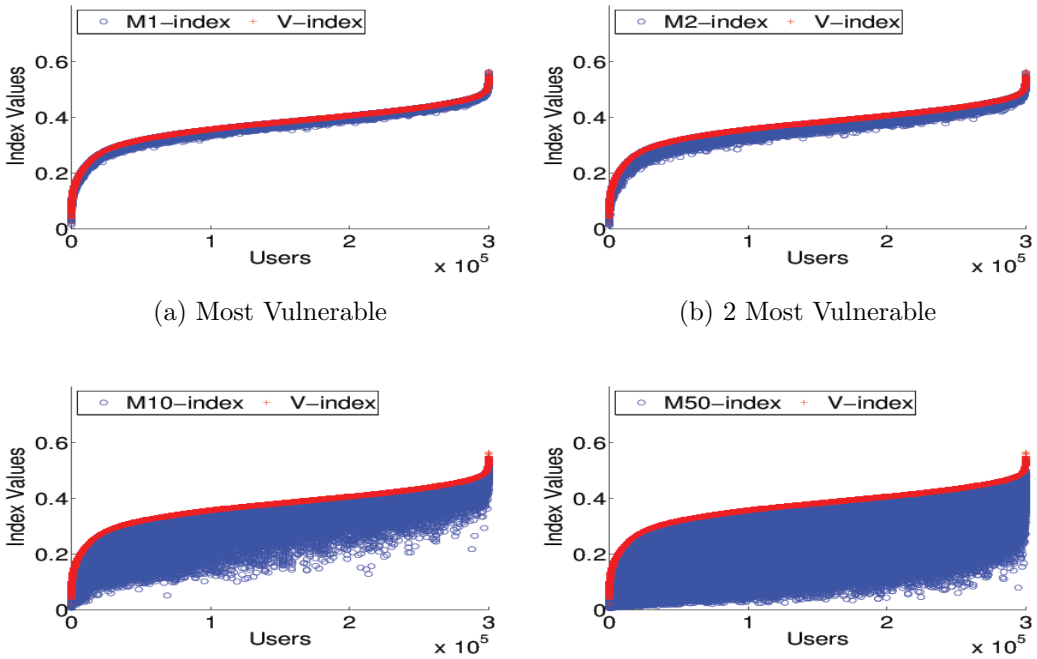
(b) C-Attributes are Visible

Figure 2.7: Impact of New Friendship (Blue) on Users with Low V-Indexes (Red) from Users with High V-Indexes

2.4.3 Vulnerability Reduction with Social Utility Constraints

To evaluate theoretical findings on vulnerability reduction with social utility constraints, I again design three experiments. First, in Figure 2.8, I compare the V-index value of each user before and after the unfriending of k most vulnerable friends. I refer this as the baseline. I do not consider social utility loss constraint when obtaining the baseline. Second, in Figure 2.9, I compare the V-index value of each user before and after the unfriending of at most k vulnerable friends while maintaining the acceptable level of social utility loss. I refer this approach as a social utility loss based approach. Third, in Figure 2.10, I compare the reductions of the baseline and social utility loss approach for each user. I observe from these experiments that it is possible to suggest the maximum number of vulnerable friends to unfriend to achieve desired user vulnerability reduction while maintaining an acceptable level of social utility loss.

For each graph in Figures 2.8, 2.9, and 2.10, X-axis and Y-axis indicate users and their V-index values, respectively. Without loss of generality, I sort all users, in the ascending order based on the existing V-index values, before I plot the graphs in



(a) Most Vulnerable

(b) 2 Most Vulnerable

(c) 10 Most Vulnerable

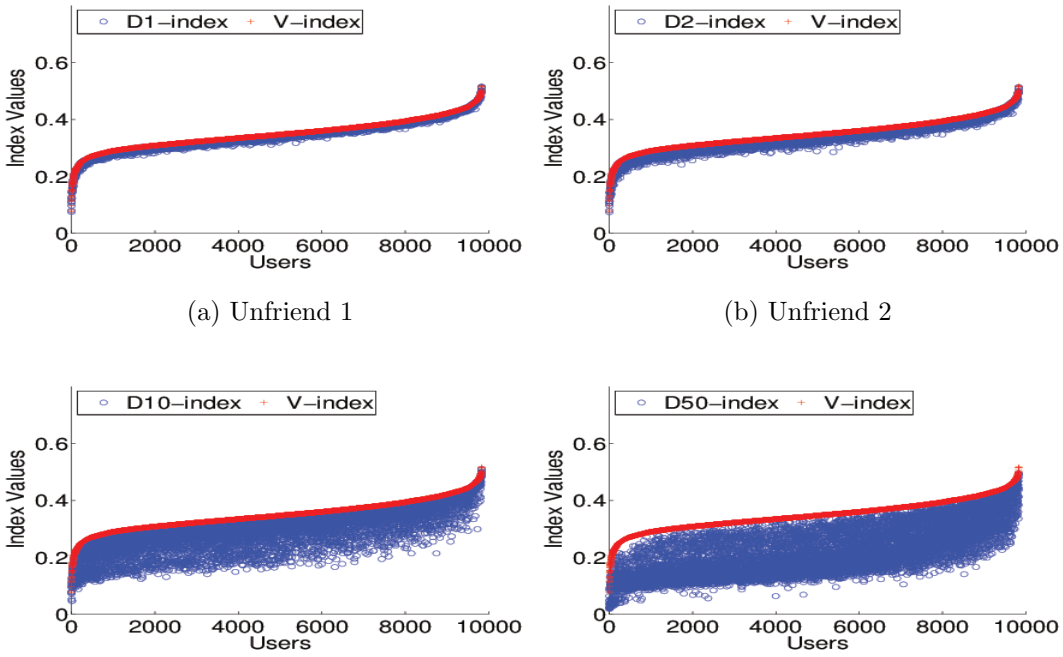
(d) 50 Most Vulnerable

Figure 2.8: (The Baseline) Performance Comparison of V-Index Values for Each User Before (Red) and After (Blue) Unfriending the k Most Vulnerable Friends from His Social Network.

Figures 2.8 and 2.9. For Figure 2.10, I sort all users in the ascending order based on the V-index values computed using the baseline approach.

The baseline

It consists of results from solving the **VMC** problem with linear objective function $\sigma(\cdot)$. As shown in Section 2.1, such a problem can be solved in polynomial time. k (or $|D_{u,0}|$, if $|D_{u,0}| < k$) most vulnerable friends are selected for removal to minimize the objective function. I run this experiment on randomly selected 300K users of the Facebook dataset. Figure 2.8 shows the performance comparison of V-index values for each user before (red) and after (blue) unfriending at most k vulnerable friends. I



(a) Unfriend 1

(b) Unfriend 2

(c) Unfriend 10

(d) Unfriend 50

Figure 2.9: Performance Comparisons of V-Index Values for Each User Before (Red) and After (Blue) Unfriending at Most k Vulnerable Friends with the Total Degree of Vulnerable Users as a Social Utility Constraint. I Set an Error Parameter $\epsilon = 0.1$ (Input Parameter to FPTAS) and Retain At Least 90% of the Total Degrees of All the Vulnerable Friends After Unfriending.

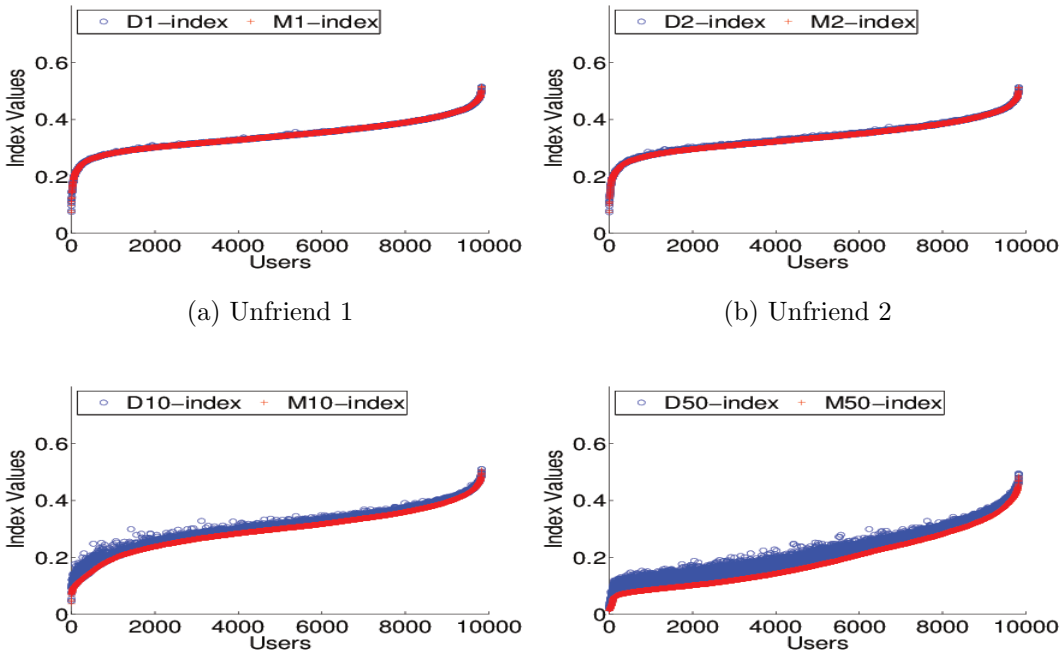
run the experiments for different values of k including 1, 2, 10, and 50. As expected, vulnerability decreases consistently as the value of k increases as seen in Figures 2.8a-2.8d. For a given k, the baseline is expected to achieve maximum user vulnerability reduction, but cannot guarantee the retention of socially valuable vulnerable friends.

A social utility based approach

I compute the minimized user vulnerability by solving the **VMUC** problem. For experiments, the V-index for each user is estimated as an average of all the P-index values of crawled friends. Hence, the objective function $\sigma(\cdot)$ for **VMUC** is linear. As discussed in Section 2.3, the scaling and rounding algorithm presented is FPTAS for such a relaxed **VMUC** problem. Figure 2.9 shows the performance comparisons of minimized V-index values for each user before (red) and after (blue) unfriending at most k vulnerable friends with the sum of all their degrees as a social utility constraint. Due to the theoretical guarantees of FPTAS, I set error parameter ϵ to relatively low value and aim to retain as high number of valuable friends as possible. I set error parameter $\epsilon = 0.1$ (input parameter to FPTAS) and retain at least 90% of the total degrees of all the vulnerable friends after unfriending. As FPTAS runs slower for a smaller error parameter, I run the experiments for randomly selected 10K users out of 300K users. As expected, vulnerability drops consistently as the value of k increases. I show the experiment results for four different k values including 1, 2, 10, and 50 in Figures 2.9a- 2.9d. I also run the experiments with other forms of social utility measures such as tie strength and number of common friends. Tie strength between two friends follows a random distribution by having a value between 0 to 1 for each user, where 1 represents the maximum social tie strength. I observe the similar patterns in user vulnerability reduction for these two social utility measures.

Comparing the social utility approach with the baseline

The purpose of this experiment is to evaluate how effective the social utility approach is in reducing vulnerability and retaining social utility. Figure 2.10 shows the performance comparison. The results for four different k values (1, 2, 10, and 50) are



(a) Unfriend 1

(b) Unfriend 2

(c) Unfriend 10

(d) Unfriend 50

Figure 2.10: Performance Comparison Between the Baseline and the Social Utility Approach.

depicted in Figures 2.10a- 2.10d. I use the total number of degrees of vulnerable friends as a social utility measure. As expected, vulnerability reduction for the baseline is more than the social utility approach. However, we can still achieve significant reduction with the social utility constraint. I observe similar results for the other two social utility measures, i.e., tie strength and number of common friends.

Before unfriending any vulnerable friend, the average V-index value of all users is 0.3485. Table 2.5 shows the average V-index values for the baseline and social utility approach. The results for the three different social utility measures are presented. It also reports the percentage decrease in average V-index value for each approach. The marginal reduction in the average V-index decreases as the value of k increases. Results show that baseline is always the most effective in removing the vulnerable

Approach	Unfriending k Friends				
	$k = 1$	$k = 2$	$k = 5$	$k = 10$	$k = 50$
Baseline	0.3425 (↓ 1.71%)	0.3371 (↓ 3.26%)	0.3216 (↓ 7.74%)	0.2944 (↓ 15.52%)	0.1891 (↓ 45.73%)
Degree	0.3427 (↓ 1.67%)	0.3381 (↓ 2.98%)	0.3261 (↓ 6.45%)	0.3048 (↓ 12.54%)	0.2185 (↓ 37.31%)
Priority	0.3426 (↓ 1.68%)	0.3378 (↓ 3.07%)	0.3250 (↓ 6.77%)	0.3025 (↓ 13.20%)	0.2146 (↓ 38.43%)
Common Friends	0.3426 (↓ 1.68%)	0.3378 (↓ 3.08%)	0.3249 (↓ 6.8%)	0.3027 (↓ 13.13%)	0.2151 (↓ 38.28%)

Table 2.5: The Average V-index Values for the Baseline and Three Different Social Utility Measures. Numbers in Brackets are Percentage Decreases in Each Average V-index Value. The Average V-index Value of All Users Before Unfriending Any Vulnerable Friend is 0.3485.

friends. This is because it removes the most vulnerable friends without considering the social utility loss for a given k. But this may incur the cost of loss in social utility value for a user. The social utility approach aims to retain the 90% of user u's social utility value for a given k in when minimizing u's vulnerability. Table 2.5 provides a summary of comparative results for $k = 1, 2, 5, 10, 50$. I observe the following: (1) the more vulnerable friends to remove, the less vulnerable a user is for all four cases (the baseline plus 3 social utility measures); (2) by allowing for a 10% loss of a social utility measure, one can still achieve comparable reduction to the baseline; (3) vulnerability reduction by all three social utility measures are similar; (4) for a given k, the baseline

achieves the largest reduction; and (5) but the gain over a social utility measure can be easily eliminated by removing the next larger k friends. For example, unfriending 2 vulnerable friends with any social utility measure can attain vulnerability reduction that is larger than that of the baseline with $k = 1$.

2.5 Summary

There are vulnerable friends on social networking sites and it is important to find and unfriend vulnerable friends so that users can improve their privacy and security. However, unfriending vulnerable friends from a user's social network can significantly decrease the user's social utility. In this chapter, I studied the novel problem of vulnerability reduction with and without social utility loss constraints. First, I provided general model for vulnerability reduction. Using this model, I formulated the two discrete optimization problems, viz., **VMC** and **VMUC**. The **VMC** problem only considers the cardinality constraint while the **VMUC** problem considers cardinality as well as social utility constraints. Both problems are NP-hard. Proposed experiments on the Facebook dataset evaluate the effectiveness of different methods of vulnerability reduction with and without social utility constraints.

I proposed a feasible approach to a novel problem of identifying a user's vulnerable friends on a social networking site. This work differs from existing work addressing social networking privacy by introducing a vulnerability-centered approach to a user security on a social networking site. On most social networking sites, privacy related efforts have been concentrated on protecting individual attributes only. However, users are often vulnerable through community attributes. Unfriending vulnerable friends can help protect users against the security risks. Based on this study of over 2 million users, I find that users are either not careful or not aware of security and privacy concerns of their friends. The proposed model clearly highlights the impact

of each new friend on a user's privacy. The proposed approach does not require the structural change of a social networking site and aims to maximally reduce a user's vulnerability while minimizing his social utility loss. The work formulates a novel problem of constrained vulnerability reduction suggests a feasible approach, and demonstrates that the problem of constrained vulnerability reduction is solvable.

Chapter 3

IDENTIFYING VULNERABILITY OF ACTIVE USERS WITH PRIVATE SETTINGS

In this chapter, my focus is on active users which mark majority of their profile settings private (i.e., Q2 users). For the rest of this chapter, they are simply referred as users, unless otherwise stated. Social media wants their users to be more social at the same time less concerned about unwarranted access to their personal data. Recent social media advancements are creating new opportunities for meaningful interactions among users, while enabling new profile settings for users to better protect their personal information. New mechanisms such as Facebook page allow users' to interact through posts without requiring them to be friends, while keeping their personal information, including demographic profiles, lists of friends, and interactions with friends private. Users' interactions on these pages are often centrally administered and publicly available for everyone. Based on whether a user can control the visibility of her actions, a post can be categorized into two parts: *personal or public post*. A personal post is a post which can be controlled by a user's individual profile settings, otherwise it is referred as a public post. In this chapter, I exclusively focus on public posts, and the users' actions, including liking, commenting and sharing, on public posts are together referred as their public interactions while the interactions on personal posts are personal interactions. Given the pervasive availability of public interactions, I ask – *are users' personal attributes predictable using only the interactions on public posts?*

To answer the question, I study the problem of the predictability of users' personal attributes in the context of Facebook pages. There are several challenges regarding

the data in Facebook pages. The first challenge is about the unavailability of users' connections. Based on the literature, users' connections play a vital role in predicting personal attributes. However, users' connections can be marked invisible using profile settings. Also, Facebook pages do not require users to form any kind of connection to interact with each other. The second challenge is about the text complexity. During the recent events across the globe including "Arab Spring", "Assam riots", and "Bangladesh Protests", Facebook users primarily communicated in their local languages such as Arabic, Assamese, and Bengali rather than English, which makes text data complex to analyze. Although interactions are pervasively available, however, public posts are visible to the public and all users can perform actions on them. This property of public posts results in the public interaction data extremely sparse and further exacerbates the difficulty of the prediction problem.

3.1 Problem Formulation

I first present the notations used in this chapter. Let $\mathbf{A} \in \mathbb{R}^{n \times m}$ be the matrix, where n is the number of rows and m is the number of columns. The entry at i-th row and j-th column of $\mathbf{A}$ is denoted as $\mathbf{A}(i, j)$. $\mathbf{A}(i, :)$ and $\mathbf{A}(:, j)$ denote the i-th row and j-th column of $\mathbf{A}$, respectively. $\|\mathbf{A}\|_F$ is the Frobenius norm of $\mathbf{A}$, and $\|\mathbf{A}\|_F = \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{m} \mathbf{A}(i, j)^2}$.

Typically, two types of objects are involved in public interactions, i.e., users and public posts in Facebook pages. Let $\mathbf{u} = \{u_1, u_2, \ldots, u_n\}$ be the set of users, and $\mathbf{v} = \{v_1, v_2, \ldots, v_m\}$ be the set of public posts, where n and m are the total numbers of users and public posts, respectively. Depending on the social media site, users' interactions involve different types of actions. For example, Facebook users mainly perform three types of actions on public posts and their associated items including liking, commenting, and sharing. For each publicly known action, I can construct the

user-post action matrix $\mathbf{R} \in \mathbb{R}^{n \times m}$, where $\mathbf{R}_{ij} = 1$ if i-th user's perform the action to j-th post, otherwise 0. For the simplicity of discussion, I assume that $\mathbf{R}$ contains user-post like actions.

Users can control their personal data such as demographic profiles and personal posts including status updates, photos and videos via their profile settings to avoid unwarranted access. However, users' interactions in Facebook pages are beyond their control and are always available to the public. Therefore in this chapter, I ask - *are users' personal attributes predictable using public interactions in Facebook pages*? To answer this question, I design a task of predicting users' personal attributes with public interaction data.

The problem of predicting users' attributes is extensively studied. It assumes that there are N users in $\mathbf{u}$ labeled with $N < n$. I assume that $\mathbf{u}_L = \{u_1, u_2, \ldots, u_N\}$ is a set of labeled users where $\mathbf{u}_L$ is a subset of $\mathbf{u}$. Let $\mathbf{Y}_L \in \mathbb{R}^{N \times K}$ be the label matrix of $\mathbf{u}_L$ where K is the total number values of a given attribute. The vast majority of existing attribute prediction algorithms make use of users' personal data such as their personal posts [62, 13] or their social networks [39, 53, 74] to obtain a predictor f to predict the attribute of users in $\{\mathbf{u} \setminus \mathbf{u}_L\}$. To seek an answer to the question of whether users' personal attributes are predictable using only public interaction data, I investigate the predictability of user's attribute with users' interactions to public posts, which is formally stated as - *Given users' public interactions on posts, the known attribute labels* $\mathbf{Y}_L$*, I aim to learn a predictor* f *to automatically predict the attribute for unlabeled users i.e,* $\{\mathbf{u} \setminus \mathbf{u}_L\}$.

3.2 Framework for Attribute Prediction: SCOUT

A user usually performs like actions with a small proportions of personal posts, resulting in a sparse user-post action relationships. One of the key difference between

public and personal posts is that only friends can perform interactions on personal posts, whereas all users can perform interactions on public posts. Hence, interaction patterns on public posts are likely to be more sparse than personal posts. Thus, the problem of predicting the personal attributes from such sparse public interactions is more challenging for traditional classification methods including support vector machines (SVM), logistic regression, and naive bayes. The proposed framework, SCOUT, aims to address the sparse interactions problem by learning a compact representation of users with the help of social theories. This compact representation is later used to build a predictor f to automatically predict the personal attributes.

3.2.1 Learning a Compact Representation

The low-rank matrix factorization-based method is one of the popular way to obtain the compact representation of users [71]. In this chapter, I adopt the well known matrix factorization model [18] to obtain low rank representation of users. The matrix factorization model seeks a low rank representation $\mathbf{U} \in \mathbb{R}^{n \times d}$ with $d << n$ via solving following optimization problem.

$$\min_{\mathbf{U},\mathbf{H},\mathbf{V}} \quad \|\mathbf{R} - \mathbf{U}\mathbf{H}\mathbf{V}^\top\|_F^2, \tag{3.1}$$

where $\mathbf{V} \in \mathbb{R}^{m \times d}$ is a low-rank space representation of the set of public posts; and $\mathbf{H} \in \mathbb{R}^{d \times d}$ captures the correlations between the low rank representations of users and public posts such as $\mathbf{R}(i,j) = \mathbf{U}(i,:)\mathbf{H}\mathbf{V}^\top(j,:)$. To avoid over-fitting, I add smoothness regularization on $\mathbf{U}$, $\mathbf{H}$, and $\mathbf{V}$ into Eq. (3.1), and then I have,

$$\min_{\mathbf{U},\mathbf{H},\mathbf{V}} \quad \|(\mathbf{R} - \mathbf{U}\mathbf{H}\mathbf{V}^\top)\|_F^2 + \lambda(\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2 + \|\mathbf{H}\|_F^2), \tag{3.2}$$

where λ is non-negative and are introduced to control the capability of $\mathbf{U}$, $\mathbf{V}$ and $\mathbf{H}$. The learnt compact representation may be inaccurate because of the sparsity of

$\mathbf{W}$. The number of zero entities in $\mathbf{R}$ is much larger than that of non-zero numbers, which indicates that $\mathbf{U}(i,:)\mathbf{H}\mathbf{V}^\top(j,:)$ will fit to be zero. The extreme sparsity of $\mathbf{R}$ will result in the learnt representation $\mathbf{U}$ close to a zero matrix.

One way to mitigate the data sparsity challenge is to give different weights to the observed and missing actions. In detail, I introduce a weight matrix $\mathbf{W} \in \mathbb{R}^{n \times m}$ where $\mathbf{W}_{ij}$ is the weight to indicate the importance of $\mathbf{R}_{ij}$ in the factorization process. The new formulation is presented in Eq. (3.2) as

$$\min_{\mathbf{U},\mathbf{H},\mathbf{V}} \quad \|\mathbf{W} \odot (\mathbf{R} - \mathbf{U}\mathbf{H}\mathbf{V}^\top)\|_F^2 + \lambda(\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2 + \|\mathbf{H}\|_F^2), \tag{3.3}$$

where $\odot$ is the Hadamard product and $(\mathbf{A} \odot \mathbf{B})(i,j) = \mathbf{A}(i,j) \times \mathbf{B}(i,j)$ for any two matrices $\mathbf{A}$ and $\mathbf{B}$ with the same size. $\mathbf{W}(i,j) = 1$ if $\mathbf{R}(i,j) = 1$. Following the suggestions in [72], I set $\mathbf{W}(i,j)$ to a small value close to zero when $\mathbf{R}(i,j) = 0$, which allows negative samples in the learning process. In this work, I set $\mathbf{W}(i,j) = 0.01$ when $\mathbf{R}(i,j) = 0$.

In addition to like actions, users can perform other actions such as sharing and commenting. There are many social theories such as homophily [52] and consistency [1] theories developed to explain users' actions. These social theories pave a way for us to model user-user and post-post correlations, which can potentially further mitigate the data sparse problem.

3.2.2 Modeling Correlations

User-user and post-post correlations in social media are widely used to improve various tasks such as feature selection [73], sentiment analysis [38, 45] and recommendation [49]. Next, I propose a novel way to compute the user-user and post-post correlations to tackle the sparsity problem further using users' actions on public posts and their associated items such as comments and shared posts.

Modeling User-User Correlations

Apart from likes, users also perform other actions including commenting, replying and sharing on different types of objects such as posts, shared posts, and comments. This subsection provides a way to include these users' activities by modeling user-user correlations. Homophily [52] is one of the important social theories developed to explain users' actions during interactions in the real world. Homophily theory suggests that similar users are likely to perform similar actions. These intuitions motivate us to obtain low-rank space representation of users based on their historical actions during interactions. I define $\Psi(i, j)$ to measure the user-user correlation coefficients between u_i and u_j. There are many ways to measure user-user correlation, such as similarity of users' behavior [50] and connections in social networks [49]. In this chapter, I choose the similarity of users' historical behavior to measure user-user correlations. A user can perform a variety of actions, including liking, commenting, and sharing. Hence, similarity is calculated as a function of the total amount of actions performed by two users together:

$$\Psi(i, j) = h(l(i, j),\ c(i, j),\ s(i, j)), \tag{3.4}$$

where $l(i, j)$, $c(i, j)$ and $s(i, j)$ record the number of likes, comments and shares, respectively, performed by u_i and u_j together. $h(\cdot)$ combines these users' behaviors together, which is defined as a sign function in this chapter:

$$\Psi(i, j) = \begin{cases} 1 & \text{if } l(i, j) + c(i, j) + s(i, j) > 0, \\ 0 & \text{Otherwise.} \end{cases} \tag{3.5}$$

With $\Psi(i, j)$, I model user-user correlations by minimizing the following term as

$$\min \quad \sum_{i=1}^{n} \sum_{j=1}^{n} \Psi(i, j) \|\mathbf{U}(i, :) - \mathbf{U}(j, :)\|_2^2 \tag{3.6}$$

Users close to each other in the low-rank space are more likely to be similar and their distances in the latent space are controlled by their correlation coefficients. For example, $\Psi(i,j)$ controls the latent distance between u_i and u_j. A larger value of $\Psi(i,j)$ indicates that u_i and u_j are more likely to be similar. Thus, I force their latent representation should be as close as possible, while a smaller value of $\Psi(i,j)$ tells that the distance of their latent representation should be larger.

For a particular user u_i, the terms in Eq. (3.6) related to her latent representation $\mathbf{U}_i$ are,

$$\min \quad \sum_{j=1}^{n} \Psi(i,j) \|\mathbf{U}(i,:) - \mathbf{U}(j,:)\|_2^2 \tag{3.7}$$

I can see that the latent representation of u_i is smoothed with other users, controlled by $\Psi(i,j)$, hence even for long tail users, with a few or even without any actions, I can still get an approximate estimate of their latent representation via user-user correlations, addressing the sparsity problem in Eq. (3.3).

After some derivations, I can get the matrix form of Eq. (3.6) as

$$\begin{aligned}
&\frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \Psi(i,j) \|\mathbf{U}(i,:) - \mathbf{U}(j,:)\|_2^2 \\
&= \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{d} \Psi(i,j) \big(\mathbf{U}(i,k) - \mathbf{U}(j,k)\big)^2 \\
&= \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{d} \Psi(i,j) \mathbf{U}^2(i,k) - \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{d} \Psi(i,j) \mathbf{U}(i,k) \mathbf{U}(j,k) \\
&= \sum_{k=1}^{d} \mathbf{U}^\top(:,k) (\mathbf{D}^u - \mathbf{S}) \mathbf{U}(:,k) \\
&= Tr(\mathbf{U}^\top \mathcal{L}^u \mathbf{U}),
\end{aligned} \tag{3.8}$$

where $\mathcal{L}^u = \mathbf{D}^u - \mathbf{S}$ is the Laplacian matrix and $\mathbf{D}^u$ is a diagonal matrix with the i-th diagonal element $\mathbf{D}^u(i,i) = \sum_{j=1}^{n} \Psi(j,i)$. $\mathbf{S}$ is the user-user correlation matrix

defined as

$$\mathbf{S} = \begin{pmatrix} \Psi(1,1) & \Psi(1,2) & \cdots & \Psi(1,n) \\ \Psi(2,1) & \Psi(2,2) & \cdots & \Psi(2,n) \\ \vdots & \vdots & \ddots & \vdots \\ \Psi(n,1) & \Psi(n,2) & \cdots & \Psi(n,n) \end{pmatrix}$$

Modeling Post-Post Correlations

Apart from likes, posts also receive other actions including commenting and sharing from users. This subsection provides a way to include these activities on posts. Consistency [1] is one of the important social theories developed to explain users' actions, which suggests that users' actions on similar posts are likely to remain consistent. These intuitions motivate us to obtain low-rank space representation of posts based on historical actions received by them. I define $\Phi(i,j)$ to measure the post-post correlation between v_i and v_j. In this chapter, I choose the similarity of actions received by posts to measure post-post correlations. A post can receive a variety of actions, including liking, commenting, and sharing. Hence, similarity is calculated as a function of the total amount of actions received by two posts together:

$$\Phi(i,j) = g(l(i,j),\ c(i,j),\ s(i,j)), \tag{3.9}$$

where $l(i,j)$, $c(i,j)$ and $s(i,j)$ record the number of users who perform likes, comments and shares, respectively, on p_i and p_j together. $g(\cdot)$ combines these users' behaviors together, which is defined as a sign function in this chapter:

$$\Phi(i,j) = \begin{cases} 1 & \text{if } l(i,j) + c(i,j) + s(i,j) > 0, \\ 0 & \text{Otherwise.} \end{cases} \tag{3.10}$$

With $\Phi(i,j)$, I model post-post correlations by minimizing the following term as

$$\min \quad \sum_{i=1}^{m}\sum_{j=1}^{m}\Phi(i,j)\|\mathbf{V}(i,:)-\mathbf{V}(j,:)\|_2^2 \tag{3.11}$$

Posts close to each other in the low-rank space are more likely to be similar and their distances in the latent space are controlled by their correlation coefficients. For example, $\Phi(i,j)$ controls the latent distance between v_i and v_j. A larger value of $\Phi(i,j)$ indicates that v_i and v_j are more likely to be similar. Thus, I force their latent representations should be as close as possible, while a smaller value of $\Phi(i,j)$ tells that the distance of their latent representation should be larger.

For a particular post v_i, the terms in Eq. (3.6) related to its latent representation $\mathbf{V}_i$ are,

$$\min \quad \sum_{j=1}^{m}\Phi(i,j)\|\mathbf{V}(i,:)-\mathbf{V}(j,:)\|_2^2 \tag{3.12}$$

I can see that the latent representation of v_i is smoothed with other posts, controlled by $\Phi(i,j)$, hence even for long tail posts, with a few or even without any actions, I can still get an approximate estimate of their latent representation via post-post correlations, addressing the sparsity problem in Eq. (3.3).

By following the derivations in Eq. (3.8), I can get the matrix form of Eq. (3.11) as

$$\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\Phi(i,j)\|\mathbf{V}(i,:)-\mathbf{V}(j,:)\|_2^2 = Tr(\mathbf{V}^\top\mathcal{L}^v\mathbf{V}), \tag{3.13}$$

where $\mathcal{L}^v = \mathbf{D}^v - \mathbf{P}$ is the Laplacian matrix and $\mathbf{D}^v$ is a diagonal matrix with the i-th diagonal element $\mathbf{D}^v(i,i) = \sum_{j=1}^{m}\Phi(j,i)$. $\mathbf{P}$ is the post-post correlation matrix defined as

$$\mathbf{P} = \begin{pmatrix} \Phi(1,1) & \Phi(1,2) & \cdots & \Phi(1,n) \\ \Phi(2,1) & \Phi(2,2) & \cdots & \Phi(2,n) \\ \vdots & \vdots & \ddots & \vdots \\ \Phi(n,1) & \Phi(n,2) & \cdots & \Phi(n,n) \end{pmatrix}$$

3.2.3 The Proposed Algorithm to Learn Compact Representation

With the components of modeling user-user and post-post correlations, the proposed algorithm is to solve the following optimization problem first.

$$\begin{aligned} \min_{\mathbf{U},\mathbf{H},\mathbf{V}} \quad & \|\mathbf{W} \odot (\mathbf{R} - \mathbf{U}\mathbf{H}\mathbf{V}^\top)\|_F^2 + \lambda(\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2 + \|\mathbf{H}\|_F^2) \\ & + \alpha Tr(\mathbf{U}^\top \mathcal{L}_u \mathbf{U}) + \beta Tr(\mathbf{V}^\top \mathcal{L}_v \mathbf{V}), \end{aligned} \tag{3.14}$$

where the first term is used to exploit the available users' like actions on posts, second term captures user-user correlations, and post-post correlations are captured by third term. The parameter α and β is introduced to control the contribution from user-user and post-post correlations, respectively.

The optimization problem in Eq. (3.14) is a multi-objective with respect to the three variables $\mathbf{U}$, $\mathbf{H}$, and $\mathbf{V}$ together. A local minimum of the objective function $\mathcal{J}$ in Eq. (3.14) can be obtained through an alternative scheme.

Computation of H. Optimizing the objective function in Eq. (3.14) with respect to $\mathbf{H}$ is equivalent to solving

$$\min_{\mathbf{H}} \quad \mathcal{J}_H = \|\mathbf{W} \odot (\mathbf{R} - \mathbf{U}\mathbf{H}\mathbf{V}^\top)\|_F^2 + \lambda\|\mathbf{H}\|_F^2 \tag{3.15}$$

The derivative of $\mathcal{J}_H$ with respect to $\mathbf{H}$ is

$$\frac{d\mathcal{J}_H}{d\mathbf{H}} = -2\mathbf{U}^\top(\mathbf{W} \odot \mathbf{W} \odot \mathbf{R})\mathbf{V} + 2\mathbf{U}^\top(\mathbf{W} \odot \mathbf{W} \odot \mathbf{U}\mathbf{H}\mathbf{V}^\top)\mathbf{V} + 2\lambda\mathbf{H} \tag{3.16}$$

Hence, the minimum can be achieved by substituting

$\left[\frac{d\mathcal{J}_H}{d\mathbf{H}}\right](i,j) = 0$. Thus, I obtain

$$[-\mathbf{U}^\top(\mathbf{W}\odot\mathbf{W}\odot\mathbf{R})\mathbf{V} + \mathbf{U}^\top(\mathbf{W}\odot\mathbf{W}\odot\mathbf{U}\mathbf{H}\mathbf{V}^\top)\mathbf{V} + \lambda\mathbf{H}](i,j) = 0 \tag{3.17}$$

Similar to [18], it leads to the updating rule of $\mathbf{H}$,

$$\mathbf{H}(i,j) \leftarrow \mathbf{H}(i,j)\sqrt{\frac{[\mathbf{U}^\top(\mathbf{W}\odot\mathbf{W}\odot\mathbf{R})\mathbf{V}](i,j)}{[\mathbf{U}^\top(\mathbf{W}\odot\mathbf{W}\odot\mathbf{U}\mathbf{H}\mathbf{V}^\top)\mathbf{V} + \lambda\mathbf{H}](i,j)}} \tag{3.18}$$

Computation of U. Similar to the computation of $\mathbf{H}$, optimizing the objective function in Eq. (3.14) with respect to $\mathbf{U}$ leads to the updating rule of $\mathbf{U}$,

$$\mathbf{U}(i,j) \leftarrow \mathbf{U}(i,j)\sqrt{\frac{[(\mathbf{W}\odot\mathbf{W}\odot\mathbf{R})\mathbf{V}\mathbf{H}^\top + \alpha\mathbf{S}\mathbf{U}](i,j)}{[(\mathbf{W}\odot\mathbf{W}\odot\mathbf{U}\mathbf{H}\mathbf{V}^\top)\mathbf{V}\mathbf{H}^\top + \alpha\mathbf{D}_u\mathbf{U} + \lambda\mathbf{U}](i,j)}} \tag{3.19}$$

Computation of V. Similarly, optimizing the objective function in Eq. (3.14) with respect to $\mathbf{V}$ leads to the updating rule of $\mathbf{V}$,

$$\mathbf{V}(i,j) \leftarrow \mathbf{V}(i,j)\sqrt{\frac{[(\mathbf{W}\odot\mathbf{W}\odot\mathbf{R})^\top\mathbf{U}\mathbf{H} + \beta\mathbf{P}\mathbf{V}](i,j)}{[(\mathbf{W}\odot\mathbf{W}\odot\mathbf{U}\mathbf{H}\mathbf{V}^\top)^\top\mathbf{U}\mathbf{H} + \beta\mathbf{D}_v\mathbf{V} + \lambda\mathbf{V}](i,j)}} \tag{3.20}$$

It can be proven that updating rules in Eq. (3.18), Eq. (3.19) and Eq. (3.20) are guaranteed to converge. Since the proof process is similar to that in [66, 18], to save space, I omit the detailed proof of the convergence of the updating rules in Eq. (3.18), Eq. (3.19) and Eq. (3.20).

In summary, I present the computational algorithm for optimizing Eq. (3.14) in Algorithm 1. In the algorithm, I conduct initialization of two Laplacian matrices, and random initialization of three matrices to be inferred from lines 1 and 2. T is the number of maximum iterations. The three matrices are updated with the updating rules until convergence or reaching the number of maximum iterations. The correctness and convergence of the updating rule can be proved with standard auxiliary function approach [66, 18].

Algorithm 1 The Proposed Algorithm to Learn Compact Representation

Input: $\mathbf{R}$, $\mathbf{S}$, $\mathbf{P}$, d, λ, α, β, T

Output: $\mathbf{U}$, $\mathbf{V}$

1: Construct matrices $\mathbf{L}_u$ and $\mathbf{L}_v$ in Eq. 3.8 and 3.13

2: Initialize matrices $\mathbf{U}$, $\mathbf{V}$ and $\mathbf{H}$ randomly

3: **while** Not convergent and $t \leq T$ **do**

4: Update $\mathbf{H}(i,j) \leftarrow \mathbf{H}(i,j) \sqrt{\frac{\left[\mathbf{U}^\top(\mathbf{W}\odot\mathbf{W}\odot\mathbf{R})\mathbf{V}\right](i,j)}{\left[\mathbf{U}^\top(\mathbf{W}\odot\mathbf{W}\odot\mathbf{U}\mathbf{H}\mathbf{V}^\top)\mathbf{V}+\lambda\mathbf{H}\right](i,j)}}$

5: Update $\mathbf{U}(i,j) \leftarrow \mathbf{U}(i,j) \sqrt{\frac{\left[(\mathbf{W}\odot\mathbf{W}\odot\mathbf{R})\mathbf{V}\mathbf{H}^\top+\alpha\mathbf{S}\mathbf{U}\right](i,j)}{\left[(\mathbf{W}\odot\mathbf{W}\odot\mathbf{U}\mathbf{H}\mathbf{V}^\top)\mathbf{V}\mathbf{H}^\top+\alpha\mathbf{D}_u\mathbf{U}+\lambda\mathbf{U}\right](i,j)}}$

6: Update $\mathbf{V}(i,j) \leftarrow \mathbf{V}(i,j) \sqrt{\frac{\left[(\mathbf{W}\odot\mathbf{W}\odot\mathbf{R})^\top\mathbf{U}\mathbf{H}+\beta\mathbf{P}\mathbf{V}\right](i,j)}{\left[(\mathbf{W}\odot\mathbf{W}\odot\mathbf{U}\mathbf{H}\mathbf{V}^\top)^\top\mathbf{U}\mathbf{H}+\beta\mathbf{D}_v\mathbf{V}+\lambda\mathbf{V}\right](i,j)}}$

7: $t = t + 1$

8: **end while**

Algorithm 2 The Proposed Framework SCOUT.

Input: $\mathbf{R}$, $\mathbf{S}$, $\mathbf{P}$ λ, α, β, T, and $\mathbf{Y}_L$

Output: A SVM Classifier f

1: Learn the compact representation $\mathbf{U}$ for $\mathbf{u}$ by Algorithm 1;

2: Obtain the compact representation $\mathbf{U}_L$ for the labeled users $\mathbf{u}_L$;

3: Train a SVM classifier f with $\mathbf{U}_L$ and $\mathbf{Y}_L$;

3.2.4 Building A Classifier for Attribute Prediction

After obtaining the low-rank representation of $\mathbf{U}$ by Algorithm 1, I choose the well-known linear SVM as the basic classifier for the attribute prediction task. The detailed framework SCOUT is presented in

Next I briefly review the above algorithm. In line 1, I learn the compact representation by Algorithm 1, and in line 3, I training a SVM classifier based on the

representation of the labeled users $\mathbf{U}_L$ and their labels $\mathbf{Y}_L$. Note that the proposed framework uses SVM as the basic classifier which only takes discrete values as labels such as gender, sexual orientation, relationship status, and religious affiliations etc. For a continuous valued attribute such as age, I simply perform discretization for SVM. Actually I can choose regression models as basic models to deal with continued value attributes and I would like to leave it as the future work.

3.3 Experiments

In this section, I conduct experiments to answer the following two questions - (1) can the proposed framework predict users' attributes from public interaction data? and (2) is it necessary to learn a compact representation? After the introduction of experimental settings, I design experiments to answer above two questions and finally perform analysis for the important parameters of the proposed framework.

3.3.1 Facebook Datasets

For experiments, I collect a Facebook dataset consisting of users' interactions on the "Basher Kella" Page[1] during the recent events of the Bangladesh protests[2]. The "Basher Kella" Facebook page represents the influential political organization in Bangladesh which also has the records of supporting violence[3]. This Facebook page[4] was founded on March 7, 2013. From March 7, 2013 till April 21, 2013, I collect Facebook users' actions on all the posts published on this page. The majority of

[1] `https://www.facebook.com/newbasherkella`

[2] `http://en.wikipedia.org/wiki/2013_Shahbag_protests`

[3] `http://www.thedailystar.net/beta2/news/net-instigation-in-full-force/`

[4] Old version of the Basher Kella Facebook page was banned during the recent events of the Bangladesh protests due to its violent content. On March 7, a new page was created which can be accessed using `https://www.facebook.com/newbasherkella`

these posts, comments and replies are written in Bengali language.

Table 3.1: Statistics of the Facebook Dataset

# of days crawl	47
# of users	498,674
# of public posts	9,907
Avg # of Likes per user	15.87
Avg # of Likes per post	580.10
Avg # of Comments per user	1.20
Avg # of Comments per post	44.11
Avg # of Shares per user	2.79
Avg # of Shares per post	139.89

Originally I collect 42,599 public posts and the administrators of this page contribute 23.25% (9907) of the total number of posts (admin-posts). As I expected, users' actions on the admin-posts were significantly higher than the posts from other Facebook users. Majority of the posts from other users contains no actions. Therefore I only use public posts from the administrators of the page in this study. For each post, I collect all the users who likes, comments and shares it. For each comment on a post, I collect all the users who like, and reply it. For each reply, I collect all the users who likes it. For each share, I collect all the users who like and comment on it. Finally, for each comment on a share, I also collect users who like, and reply it. Also, for each reply on a share comment, I collect all the users who likes it. Table 3.1 shows the overall statistics of the dataset.

In this work, I choose three attributes, i.e., religious affiliation, relationship status and sexual orientation. For the religious affiliation attribute, to establish the ground truth for evaluation, I first use the Facebook graph search results to examine the set of users who set the attribute available to the public and then collect the attributes of these users with their public interaction data to establish a dataset, Facebook-religion, to assess the performance of the proposed framework. The statistics of Facebook-religion is shown in Table 3.1a. I only find a small portion of users (2853 out of 42,599) who set their religious affiliation publicly available, which is consistent with the observation in [32]. I obtain 5 values for religious affiliation, i.e., *Muslim, Atheist, Buddhist, Hindu, or Christian* and these five religions are indeed the top five religions in Bangladesh based on their populations[5]. To assess the performance of the proposed framework with other attributes, I also choose another two attributes, relationship status and sexual orientation. For each of these two attributes, following a similar process, I build a dataset based on 2853 users. The statistics of Facebook-relation and Facebook-sexual are shown in 3.1b and 3.1c, respectively. For relationship status, I consider two values, *single* and *not-single*. All the Facebook users with relation status values as "married", "engaged" and "in a relationship" are considered not-single, whereas "single", "widowed", and "divorced" are considered single. Sexual orientation attributes are interpreted using "Interested In" values from Facebook users' profiles. I consider three values for sexual orientation such as users who *like men*, users who *like women*, and users who *like both men and women*.

For each dataset, I choose $x\%$ of the dataset for training and the remaining $(1-x)\%$ as testing. In this work, I vary x as $\{50, 60, 70, 80, 90\}$. For each x, I repeat the experiments 5 times and report the average performance.

From the evaluation perspective, precision and recall are equally important for

[5]`http://en.wikipedia.org/wiki/Bangladesh`

Religion	# of Users	Percentages(%)
Muslim	1866	65.40
Atheist	216	7.57
Buddhist	113	3.96
Hindu	463	16.23
Christian	195	6.84
Total	2853	100

(a) Statistics of Facbook-Religion Dataset

Values	# of Users	Percentages(%)
Single	760	65.01
Not-single	409	34.99
Total	1169	100

(b) Statistics for the Facebook-Relation Dataset

Values	# of Users	Percentages(%)
Men	196	9.65
Women	507	24.96
Men & Women	1328	65.39
Total	2031	100

(c) The Statistics for Facebook-Interested-In Dataset

the prediction task. For example, in an Islamic country like Bangladesh, the cost of incorrectly predicting someone as atheist could be disastrous, as it carries connotations of blasphemy[6]. However, precision, recall, and F1-score are biased towards one of the labels. Hence, it is unsuitable for the unbalanced evaluation dataset. For this purpose, I use commonly adopted *macro-average F1* score to assess the prediction performance, as it gives equal weight to all the labels. The macro-average F1 score is defined as,

$$macro - F1 = \frac{\sum_{i=1}^{K} F_i}{K}, \text{ where } F_i = \frac{2p_i r_i}{p_i + r_i}, \tag{3.21}$$

where p_i and r_i refer to the precision and recall values associated with the i-th label, respectively. Note that F-score can not be computed for a baseline which always picks the majority label for the prediction.

3.3.2 Performance Evaluation

In this subsection, I conduct experiments to answer the first question - can the proposed framework predict users' personal attributes from users' public interaction data? To answer this question, I investigate the performance of the proposed framework by comparing it with the random performance. For SCOUT, I choose the cross-validation to determine the parameter values and more details about the parameter analysis will be discussed in the following subsection. I empirically set the number of latent dimensions d to 50. The performance results for Facebook-religion, Facebook-relation and Facebook-sexual are demonstrated in Figures 3.1a, 3.1b and 3.1c, respectively.

I have the following observations:

- For all the Figures 3.1a, 3.1b and 3.1c, the performance of SCOUT increases

[6] `http://en.wikipedia.org/wiki/Atheism_and_religion#Atheism_in_Islam`

with the increase of x. This is due to the fact that more training data helps to build a better SVM classifier.

- The proposed framework consistently outperforms the random method. The proposed algorithm gains up to 70.49% and 49.83% relative improvement in Facebook-religion and Facebook-sexual, respectively. I conduct a t-test on these results and the evidence from t-test suggests that the improvement is significant. These results support that users' personal attributes are predictable from public interaction data. In the following subsections, I will investigate the contributions from different components to this improvement.

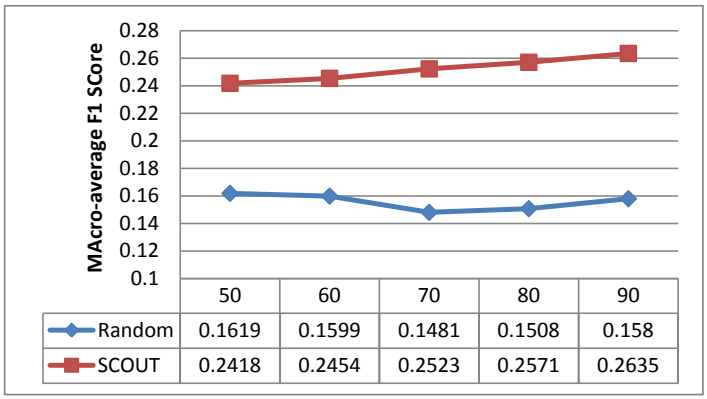
In conclusion, above results suggest a positive answer to the first questions - the proposed framework can predict various users' personal attributes from public interaction data.

3.3.3 Impact of the Learnt Compact Representation

As mentioned above, the public interaction data representation matrix $\mathbf{R}$ is very sparse and I proposed an algorithm to learn a compact representation $\mathbf{U}$ with the help of social theories. In this subsection, I study the impact of the compact representation on the performance of the proposed framework to answer the second question. In detail, I define the following variants:

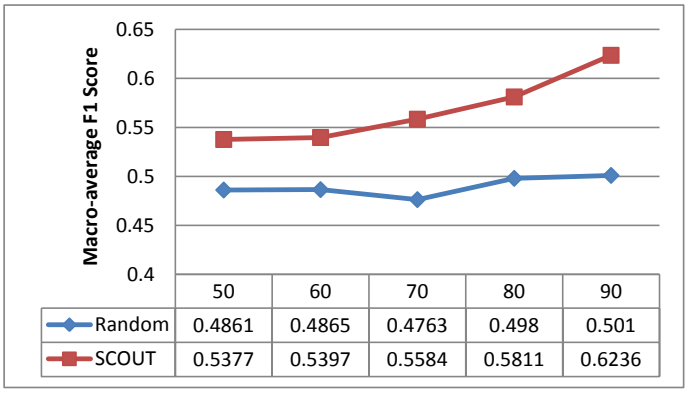
- SCOUT-Corr: Eliminate the impact of the user-user and post-post correlations by setting $\alpha = \beta = 0$ in Eq. (3.14);

- SCOUT-Corr-W: Eliminate the effects from both the correlation and the weight matrix $\mathbf{W}$ by setting $\alpha = \beta = 0$ and $\mathbf{W}$ to be a matrix with all entities equal to 1 in Eq. (3.14); and



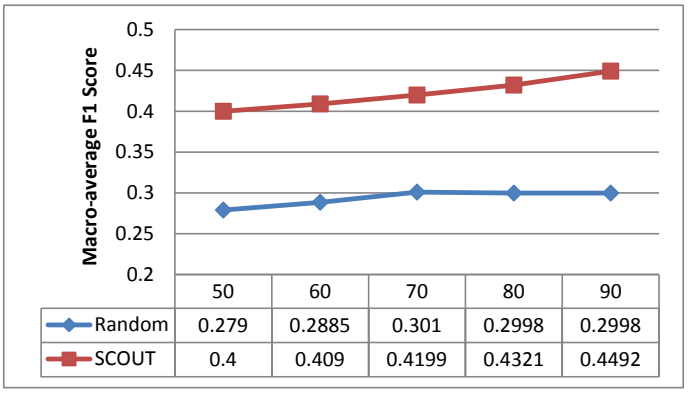
	50	60	70	80	90
Random	0.1619	0.1599	0.1481	0.1508	0.158
SCOUT	0.2418	0.2454	0.2523	0.2571	0.2635

(a) Religious Affiliation



	50	60	70	80	90
Random	0.4861	0.4865	0.4763	0.498	0.501
SCOUT	0.5377	0.5397	0.5584	0.5811	0.6236

(b) Relationship Status



	50	60	70	80	90
Random	0.279	0.2885	0.301	0.2998	0.2998
SCOUT	0.4	0.409	0.4199	0.4321	0.4492

(c) Interested In

Figure 3.1: Prediction Performance of the Proposed Framework

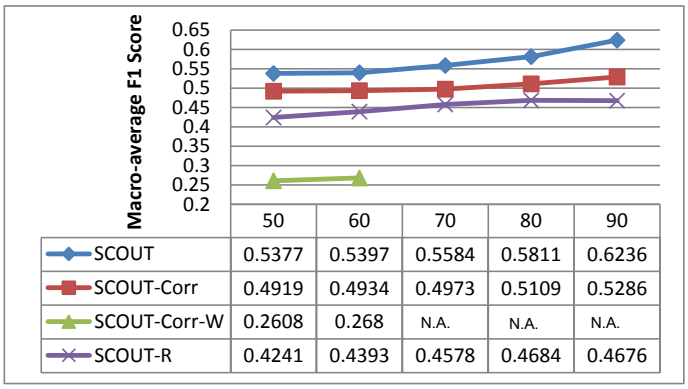
- SCOUT-R: Eliminate the impact of the compact representation by learning the SVM classifier with the original matrix $\mathbf{R}$.

I only show results on Facebook-relation in Figure 3.2 since I have similar observations with other settings. Note that sometimes the classifier gives the majority prediction and I can not compute macro-F1 in this situation; hence I use "N.A." to denote the performance in the table. When eliminating the impact of the user-user and post-post correlations, the performance of SCOUT-Corr reduces, which indicates the importance of incorporating these correlations based on social theories. When eliminating these correlations and the weight matrix $\mathbf{W}$, a traditional matrix factorization algorithm learns the compact representation and the performance of SCOUT-Corr-W reduces dramatically. As mentioned before, the extreme sparsity of $\mathbf{R}$ will lead to the learned compact representation close to zero. When building the classifier based on the sparse matrix $\mathbf{R}$, the performance of SCOUT-R also reduces a lot. I can not learn a good classifier based on the sparse and high-dimensional matrix $\mathbf{R}$, which directly supports the importance of learning a compact representation for uesrs.

In conclusion, the learned compact representation can mitigate the sparse problem of public interaction data and plays an important role in the performance improvement of the proposed framework, which correspondingly answers the section question.

3.3.4 Impact of User-User and Post-Post Correlations

The parameters α and β are introduced to control the contributions from user-user and post-post correlations for the proposed framework SCOUT. Therefore, I investigate the impact of user-user and post-post correlations via analyzing how the changes in α and β affect the performance of SCOUT in terms of the attribute prediction. I vary the value of α and β as $\{0, 0.01, 0.1, 1, 10, 100, 1000\}$. The results are shown in Figure 3.3 for Facebook-religion. I ignore the results with other settings



	50	60	70	80	90
SCOUT	0.5377	0.5397	0.5584	0.5811	0.6236
SCOUT-Corr	0.4919	0.4934	0.4973	0.5109	0.5286
SCOUT-Corr-W	0.2608	0.268	N.A.	N.A.	N.A.
SCOUT-R	0.4241	0.4393	0.4578	0.4684	0.4676

Figure 3.2: Impact of the Learnt Compact Representation on the Proposed Framework. Note that Sometimes the Classifier Gives the Majority Prediction and I can Not Compute Macro-F1 in This Situation; Hence I Use "N.A." to Denote the Performance in the Table.

since I have similar observations.

In general, with the increase of α and β, I observe similar patterns: first increasing, reaching its peak value and then degrading rapidly. These patterns can be used to determine the optimal value of α and β for SCOUT in practice. In particular it can be observed,

- When α is increased from zero, eliminating the impact of user-user correlations to SCOUT, to 0.1, the performance improves, suggesting the importance of user-user correlations in the proposed framework to mitigate the data sparsity problem.

- When β is increased from zero, eliminating the impact of post-post correlations to SCOUT, to 0.01, the performance improves, suggesting the importance of post-post correlations in the proposed framework.

- SCOUT achieves its best performance when $\alpha = 0.1$ and $\beta = 0.01$, further demonstrating the importance of user-user and post-post correlations in

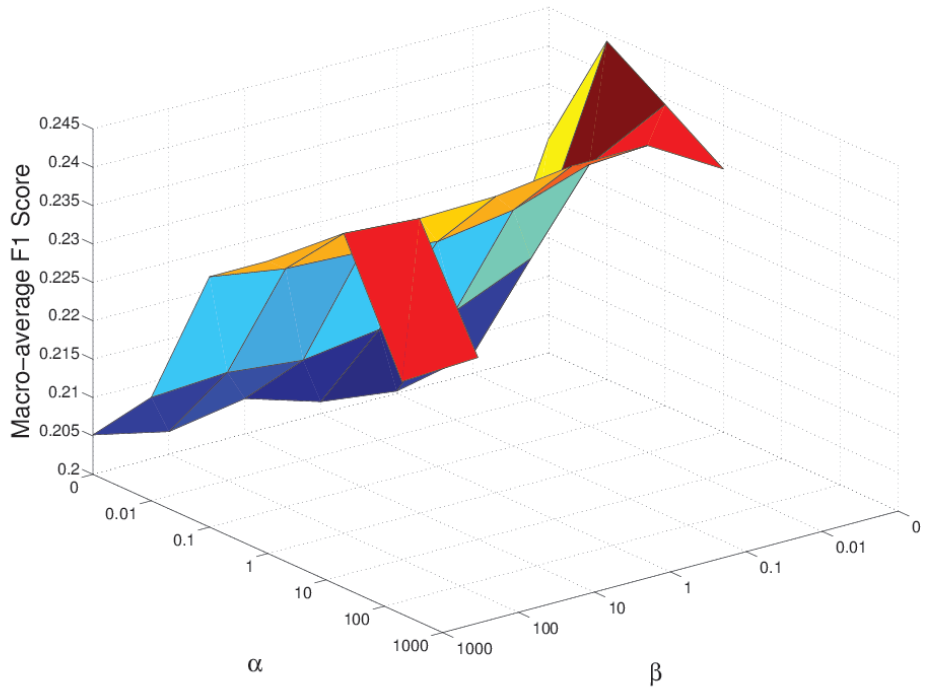


Figure 3.3: Impact of User-User and Post-Post Correlations on Predicting Religious Affiliations, Respectively.

SCOUT.

- From $\alpha = 0.1$ to $\alpha = 1000$, the performance decreases rapidly. In comparison, performance decrease is not rapid from $\beta = 0.01$ to $\beta = 1000$. When α and β is very large, user-user and post-post correlations dominate the learning process and the learned representation is inaccurate. For example, when $\alpha \to +\infty$ $\beta \to +\infty$, I will obtain a trivial solution, all $\mathbf{U}_i$ for $(1 \leq i \leq n)$ are exactly the same.

In summary appropriate incorporation of user-user and post-post correlations into the dimension reduction algorithm can greatly improve the prediction performance of personal attributes.

3.4 Summary

In order to secure users' privacy, researchers have been exploring different cues to show users vulnerabilities against privacy attacks. Unlike these methods, my focus is on predicting personal attributes from publicly available interactions alone so that users can secure privacy. To the best of my knowledge, I am the first to address this problem. I provide a way to mitigate the sparsity problem of public interaction data with the help of social theories and propose a novel framework, SCOUT, to predict users' personal attributes from public interaction data. The evaluation of the proposed framework with the real-word data from Facebook page shows that users' attributes are predictable.

Chapter 4

LITERATURE REVIEW

User privacy on a social networking site has received considerable attention recently. Gross and Acquisti [31] evaluate the amount of information disclosed through a social networking site and study usage of privacy settings. This work revealed that only a few users change the default privacy preferences on Facebook. Narayanan and Shmatikov [55, 56] show that users are not well protected on a social networking site by successfully deannonymizing network data solely based on network topology. They also highlight the fact that privacy laws are inadequate, confusing, and inconsistent amongst nations making social networking sites more vulnerable. Wondracek et al. [79] propose a simple deanonymization scheme which exploits the group membership information to breach users privacy. They also pointed out that, social network data aggregation projects such as OpenID[1], DataPortability[2], the social graph project[3], and various microformats[4] potentially represent a greater threat to an individual privacy.

Liu and Maes [46] point towards lack of privacy awareness and find large number of social network profiles in which people describe themselves using a rich vocabulary of their passions and interests. This fact strengthen the need for vulnerability research on a social networking site to make users aware of privacy risks. Krishnamurthy and Wills [42] discuss the problem of leakage of personally identifiable information and how it can be misused by third parties [56]. Ho et al. [36] discover three privacy

[1]http://openid.net/

[2]http://www.dataportability.org/

[3]http://bradfitz.com/social-graph-problem/

[4]http://microformats.org/

problems on most social networking sites. First, users are not notified by social networking sites when their personal information is at risk. Second, existing privacy protection tools are not flexible enough. Finally, users cannot prevent information that may reveal private information about themselves from being uploaded by any other user. These observations validate the index estimation method proposed in the Chapter 2.

There has been some research which suggests the fundamental changes to social networking sites to achieve user privacy. Squicciarini et al. [69] introduce a novel collective privacy mechanism for better managing the shared content between the users. Fang and LeFevre [21] focus on helping users to express simple privacy settings but they have not considered additional problems such as attribute inference [82], or shared data ownership [69]. Zheleva and Getoor [82] show how an adversary can exploit an online social network with a mixture of public and private user profiles to predict the private attributes of users. Baden et al. [6] present a framework where users dictate who may access their information and based on public-private encryption-decryption algorithms. Although the proposed framework address privacy concerns, it comes at the cost of increased response time from a social networking site. Proposed work does not suggest any fundamental changes to social networking sites. I find users can secure user privacy by unfriending the vulnerable friends. Unfriending[5] has been studied recently but I am the first one to propose unfriending to reduce the vulnerability of a user.

Psychologists have long been predicting user traits and attributes based on various types of information such as samples of written text [22], answers to psychometric tests [14], or the appearances of places people inhibit [30]. Most of these researches are based on an assumption that users have tendencies to inadvertently leave behind

[5]http://www.nytimes.com/2010/10/24/fashion/24Studied.html

cues which correlate with their personal attributes. Recently computer scientists are also exploring users' personal attributes based on cues from the web, such as user's web site browsing logs [54, 37, 17, 26], contents of personal web sites [51], and music collections [64].

Social media popularity has created several opportunities for users to create data. Massive amount of social media data has attracted attention of privacy researchers to identify, measure and mitigate the risks of predicting personal attributes [39, 53, 62, 13, 60, 28, 44, 11]. Jernigan and Mistree [39] shows that location within a friendship network at Facebook is predictive of sexual orientation. Mislove et.al. proposed a method of inferring user attributes that is inspired by examining the normalized conductance of the existing friend lists, whose value ranges from -1 to 1, with strongly positive values indicating significant community structure. Prior studies of social network graphs have found that normalized conductance values greater than 0.2 correspond to strong communities, that could be detected fairly accurately by community detection algorithms [53]. Rao et al. [63] predicts gender from tweet texts alone using an N-gram only model and hand-crafted sociolinguistic-based features. Conover et al. [12] proposed several methods for predicting the political alignment of Twitter users based on the content and structure of their political communication in the run-up to the 2010 U.S. midterm election. Quercia et al. [60] found a positive correlation between number of followers/following and age. Golbeck et al. [29] used LIWC features over a sample of 167 Facebook volunteers as well as profile information and found limited success of a regression model in predicting personality of a user. Li et al. [44] profile users' location by integrating both friendship and content information in a probabilistic model.

An inspiring work from Kosinski et.al. [41] shows that wide variety of people's highly sensitive personal attributes can be automatically and accurately inferred us-

ing the variety of Facebook likes. Personal attributes include sexual orientation, ethnicity, religious and political views, personality traits, intelligence, happiness, use of addictive substances, parental separation, age and gender. Conclusions in this work are based on the Facebook users personal data which includes their Facebook likes, detailed demographic profiles, and results of several psychometric tests. Chapter 3 assume that users' personal data is not available, as the data visibility can be controlled using users' profile settings. The focus of this Chapter is on a novel type of data which is not only publicly available but also beyond the control of users' profile settings. The proposed framework SCOUT aims to explore personal attributes from such data so that users can secure themselves against privacy attacks. Table 4 summarizes literature work on prediction of private attributes. It also shows the required data for successfully identification of those attributes. Social media related work is highlighted in **bold**.

Table 4.1: Summarization of Literature Work on Prediction of Private Attributes. Social Media Related Work is Highlighted in Bold

Private Attributes/Traits	**Required Data for Prediction**
Gender	Chat messages [43], Web-pages and their browsing history [37], Web-search query logs [40], **Linguistic features from tweets [63] and Facebook posts [61]**, Web-pages' browsing history [17, 26], **Statistical features alone from Twitter [60]**, **Full names, profiles and contents from Twitter [9]**, **Statistical and content features from Twitter [3]**, **Facebook likes [41]**, **First-names in Twitter [47]**
Age	Web-pages and their browsing history [37], Web-search query logs [40], **Linguistic features from tweets [63]**, Web-pages' browsing history [17, 26], **Statistical features alone from Twitter [60]**, **Statistical and content features from Twitter [3]**, **Facebook likes [41]**
Location	Web-search query logs [40], Visual, textual and temporal features from Flickr photos [16], GPS history data [84, 83], **Linguistic features from tweets [63]**, **Facebook network and address [5]**, **Tweet content [35]**, **Check-in data from Gowalla and Brightkite [10]**, Cell-phone location trace data [68, 10]
Relationship Status	**Facebook likes [41]**
Social Security Numbers	location and date of birth [2]

Continued on next page

Table 4.1 – *Continued from previous page*

Private Attributes/Traits	**Required Data for Prediction**
Religious views	**Facebook likes [41]**
Political views	Music Preferences [64], **Linguistic features from tweets [63]**, **Profile, network, statistical and content features from Twitter [59]**, **Twitter follower network [27]**, **Tweet content and communication network [13, 12]**, **Statistical and content features from Twitter [3]**,**Facebook likes [41]**, **Twitter retweets [80]**
Sexual orientation	**Facebook network [39]**, **Facebook likes [41]**
Occupation	Web-pages' browsing history [17]
Education level	Web-pages' browsing history [26], Web-pages' browsing history [17]
Household Income	Web-pages' browsing history [26]
Ethnicity	**Linguistic features from Facebook posts [61]**, **Profile, network, statistical and content features from Twitter [59]**, Web-pages' browsing history [26], **Facebook likes [41]**
Intelligence	**Facebook likes [41]**
Happiness	Cell-phone call logs [19], **Facebook likes [41]**
Use of addictive substance	**Facebook likes [41]**

Continued on next page

Table 4.1 – *Continued from previous page*

Private Attributes/Traits	**Required Data for Prediction**
Parental Separation	**Facebook likes [41]**
Social ties	Cell-phone call logs [19], **Photo and music related tagging data from Flickr and Last.fm [65]**, **Spatio-temporal features from Flickr photos [15]**
Size and density of the friendship network	**Facebook likes [41]**
Big Five Personality Traits (Openness, Conscientiousness, Extraversion, Agreeableness, Neuroticism)	Music Preferences [64], Personal web-pages and visitors' ratings [51], **Statistical and content features from Twitter [28]** and **Facebook profiles [29]**, **Statistical features alone from Twitter [60]**, **Facebook likes [41]**

Chapter 5

CONCLUSION AND FUTURE WORK

In this dissertation, I systematically studied how one can manage their vulnerabilities on a social networking sites. Based on profile settings and amount of available information, users are categorized into four types: (1) active users with public settings (Q1 users), (2) active users with private settings (Q2 users), (3) inactive users with public settings (Q3 users), and (4) inactive users with private settings (Q4 usres) (Please refer Figure 1.1 for details). For each of these types of users, vulnerability can be managed in three steps: (1) identifying, (2) measuring and (3) reducing a user vulnerability. The main focus of this dissertation is on active users only i.e. Q1 and Q2 users.

Chapter 2 entirely focuses on managing vulnerability of Q1 users. Since rich literature is available on identifying Q1 users vulnerabilities, this chapter provided a novel way to measure a Q1 user's vulnerability on a social networking site, and then proposed a way to mitigate its vulnerability while retaining social utility value. I proposed a feasible approach to a novel problem of identifying a user's vulnerable friends on a social networking site. This work differs from existing work addressing social networking privacy by introducing a vulnerability-centered approach to a user security on a social networking site. On most social networking sites, privacy related efforts have been concentrated on protecting individual attributes only. However, users are often vulnerable through community attributes. Unfriending vulnerable friends can help protect users against the security risks. Based on this study of over 2 million users, I find that users are either not careful or not aware of security and privacy concerns of their friends. The proposed model clearly highlights the impact

of each new friend on a user's privacy. The proposed unfriending-based mechanism does not require the structural change of a social networking site and aims to maximally reduce a user's vulnerability while minimizing his social utility loss. The work formulated a novel problem of constrained vulnerability reduction suggests a feasible approach, and demonstrated that the problem of constrained vulnerability reduction is solvable.

Chapter 3 focused on understanding vulnerabilities of Q2 users. To the best of my knowledge, little is known about the vulnerabilities of Q2 users before this dissertation work. Similar to Q1 users, the aim of this chapter is to identify the vulnerabilities of Q2 users which is achieved by predicting personal attributes from publicly available unprotected interactions alone. I provided a way to mitigate the sparsity problem of public interaction data with the help of social theories and proposed a novel framework, SCOUT, to predict users' personal attributes from public interaction data. The evaluation of the proposed framework with the real-word data from Facebook page showed that users' attributes are predictable. This work paves the way for many exiting work including measuring and mitigating vulnerabilities of Q2 users.

There are many extensions and work that are worth further explorations. I summarize the future work as below

5.1 User's Vulnerability Across Platforms

Throughout this dissertation, I focused exclusively on a single social networking platform for managing one's vulnerability. Often a user has multiple accounts on different social media sites, and is unaware of how much information about her can be publicly available for everybody. Although such information about a user can be challenging to garner at one place, it aids attribute prediction frameworks and there by making user's more vulnerable.

As a preliminary study, I designed a novel web based tool [34] for collecting attribute values of interest associated with a particular social media user. I refer to these attributes as Provenance Attributes and the tool as Provenance Data Collector. Currently this tool is designed to assist social media users to collect provenance data of more than half a billion Twitter users [1], and more than a billion Facebook users [2]. Provenance data collector [3] is an online data collection tool focusing on efficiently retrieving useful attribute values of a given Twitter or Facebook user. This tool features an intuitive user interface and is designed to enable fast retrieval of a maximum number of desired provenance attributes. If some desired provenance attributes are uncertain, the tool provides best possible URL (Uniform Resource Locator) to help users further their findings. In addition to provenance attributes, the tool also presents other attribute values and related images during the search, and measures to evaluate efficiency of the system. Figure 5.1 shows an overview of the tool for collecting provenance attribute values. The user vulnerability research can be expanded by examining the impact of such tools on a user's vulnerability. I think protecting against vulnerabilities arising from multiple platforms is a big challenge.

5.2 Exploring Inactive Users with Private Settings

Next challenge for vulnerability research is to identify the vulnerability of inactive users with private settings. There are two way to explore this problem. The first way is to exploit the unique features in the usenames alone to identify a vulnerability, whereas as other way is to obtain more information about inactive users across social media sites. Recent research has shown that often users are more active on one social

[1] `http://en.wikipedia.org/wiki/Twitter`

[2] `http://en.wikipedia.org/wiki/Facebook`

[3] The provenance data collector tool is located at `http://blogtrackers.fulton.asu.edu/Prov_Attr`, and demonstration video can be found at `http://www.screencast.com/t/XujEYbBXBKBd`

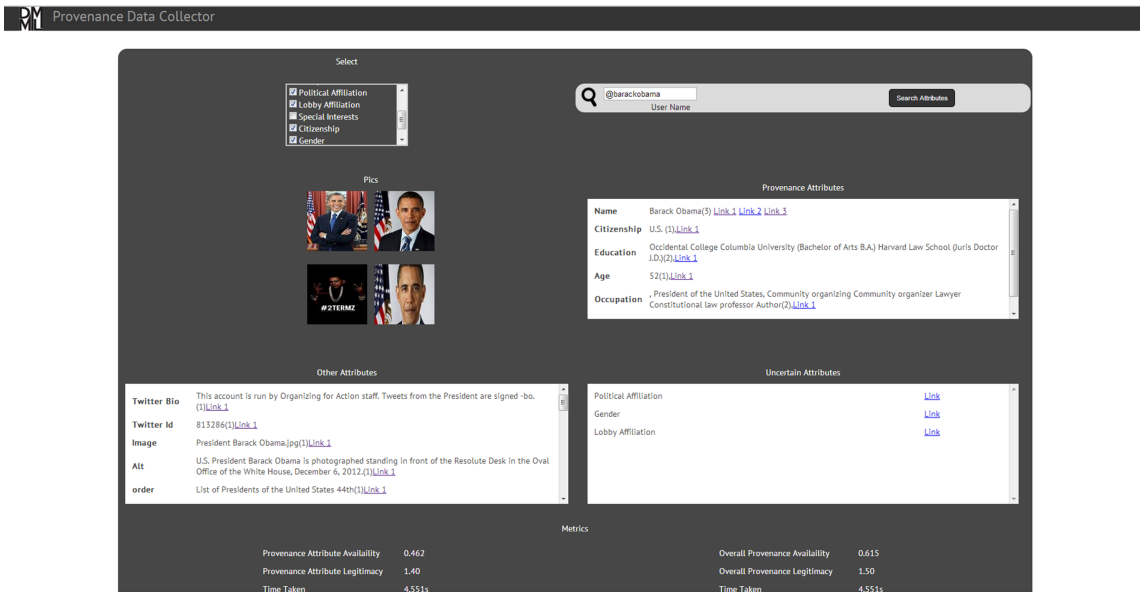


Figure 5.1: Web Interface of the Provenance Data Collector Tool Showing Attribute Values of President Barack Obama (@barackobama)

networking site than another [67]. Also, usernames alone have been used to connect unique users across social media sites [81].

5.3 Identifying Passive Attackers

Using the proposed vulnerability work, one can identify friends whose actions can potentially compromise a user privacy. Let us assume that a user is doing the most by effectively employing the profile settings to keep her personal data private. However, there still exist passive attackers who can breach a user's privacy. Passive privacy attackers are those who just want to harvest real friends to increase their credibility and used them later to target one's privacy. There are three possible cases using which a passive attacker can compromise a user's privacy: (1) a passive attacker becomes a friend with a user; (2) a passive attacker becomes a friend with a user's friend; and (3) a passive attacker is outside of a user's 2-hop network.

In the first case, a user is completely compromised as she trusts a passive attacker by becoming a friend. This case further strengthen an experimental finding that less vulnerable users can become more vulnerable if they are careless when making new friends. The proposed vulnerability approach can only be able to identify such attackers if they involved in exposing a user's private information to others. Hoiwever, proactive approaches need to be investigated and designed to mitigate the vulnerability arising from a passive attacker if he is assumed to be an on-line but not real-world friend of a user. Currently in such situation, a user can form a social circles of friends based on the ties in the on-line and real world. A user can then selectively share sensitive information among its social circles. Another approach of dealing with the first case is that behavioral patterns of friends can be investigated to automatically detect the passive privacy attackers. However, this approach will be ineffective if a passive attacker exhibit the behavioral patterns similar to those friends who are minimally exposing others.

The second case often arises when a user's friend is less careful in establishing a new friend connection. The proposed approach is more likely to identify such vulnerable friend as she is less careful about her own as well as friends privacy. A user can also notify his vulnerable friends about potential risks and request them to take necessary actions. In this case the risk of accessing a user's private information is less in comparison with the first case. From the privacy perspective, the third case is the best possible scenario in comparison with the other two. Using the proposed approach user can periodically measure vulnerability. If a user observes a rise in vulnerability, she can notify his friends about recent actions which causes the increase in her vulnerability. This approach can help a user to keep passive attacker from infiltrating 2-hop network.

Social networking sites like Facebook and GooglePlus can also help in detecting

such passive privacy attackers by analyzing their browsing (clicking) behavior [77]. Assumption here is that passive attackers are goal oriented and involved in a distinctive browsing patterns than normal users.

5.4 Extending User Vulnerability to Cope with Identity Theft

The proposed definition of a user vulnerability is based on the visibility and exposure of a users profile through not only attributes settings but also his friends. We measure a user vulnerability using three factors: (1) users privacy settings that can reveal personal information; (2) a users action on a social networking site that can expose their friends personal information; and (3) friends action on a social networking site that can reveal users personal information. In other words, the proposed vulnerability measure provides how much risk to privacy a user is due to her and friends actions.

Besides the actions of a user and friends, a user is also vulnerable to other types of privacy attacks such as identity theft or cloning. If a cloned user is involved in exposing a normal user's profile, then the proposed vulnerability measure can be able to direct to the origin of such attack. Otherwise, the proposed solution can not be able to identify a cloned user. However, a social networking site like Facebook can play a significant role in detecting a cloned user, as they are expecting to have different browsing (click) activities [77] in comparison with the normal users. Hence, further investigation needed to extend existing user vulnerability approaches to handle identity thefts or cloning attacks.

5.5 Measuring and Reducing Vulnerability of Active Users with Private Settings

In Chapter 3, I successfully demonstrate a method to predict personal attributes of active users with private settings. This paves the way to the next important

task of measuring and reducing vulnerabilities of such users. One way of reducing such vulnerabilities is by providing more control to users' so that they can control their profile settings. However such profile settings requires social networking sites to change their existing architecture and also limits users' social behavior. The other way of reducing vulnerabilities is to diversify user's activities so that the framework proposed in chapter 3 can not extract patterns and failed to predict the personal attributes.

REFERENCES

[1] R. P. Abelson. Whatever Became of Consistency Theory? *Personality and Social Psychology Bulletin*, 1983.

[2] A. Acquisti and R. Gross. Predicting social security numbers from public data. *Proceedings of the National academy of sciences*, 106(27):10975–10980, 2009.

[3] F. Al Zamal, W. Liu, and D. Ruths. Homophily and latent attribute inference: Inferring latent attributes of twitter users from neighbors. In *ICWSM*, 2012.

[4] L. Backstrom, D. Huttenlocher, J. Kleinberg, and X. Lan. Group Formation in Large Social Networks: Membership, Growth, and Evolution. In *the 12th ACM SIGKDD*, pages 44–54, 2006.

[5] L. Backstrom, E. Sun, and C. Marlow. Find me if you can: improving geographical prediction with social and spatial proximity. In *Proceedings of the 19th international conference on World wide web*, pages 61–70. ACM, 2010.

[6] R. Baden, A. Bender, N. Spring, B. Bhattacharjee, and D. Starin. Persona: An online social network with user-defined privacy. *ACM SIGCOMM Computer Communication Review*, 39(4):135–146, 2009.

[7] G. Becker. A Theory of Social Interactions, 1974.

[8] W. Brock and S. Durlauf. Discrete Choice with Social Interactions. *The Review of Economic Studies*, 68(2):235, 2001.

[9] J. D. Burger, J. Henderson, G. Kim, and G. Zarrella. Discriminating gender on twitter. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1301–1309. Association for Computational Linguistics, 2011.

[10] E. Cho, S. A. Myers, and J. Leskovec. Friendship and mobility: user movement in location-based social networks. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1082–1090. ACM, 2011.

[11] R. Cohen and D. Ruths. Classifying political orientation on twitter: Its not easy! In *Proceedings of the 7th International Conference on Weblogs and Social Media*, 2013.

[12] M. Conover, J. Ratkiewicz, M. Francisco, B. Gonçalves, F. Menczer, and A. Flammini. Political polarization on twitter. In *ICWSM*, 2011.

[13] M. D. Conover, B. Gonçalves, J. Ratkiewicz, A. Flammini, and F. Menczer. Predicting the political alignment of twitter users. In *Privacy, security, risk and trust (passat), 2011 ieee third international conference on and 2011 ieee third international conference on social computing (socialcom)*, pages 192–199. IEEE, 2011.

[14] P. T. Costa and R. R. McCrae. *Revised neo personality inventory (neo pi-r) and neo five-factor inventory (neo-ffi)*, volume 101. Psychological Assessment Resources Odessa, FL, 1992.

[15] D. J. Crandall, L. Backstrom, D. Cosley, S. Suri, D. Huttenlocher, and J. Kleinberg. Inferring social ties from geographic coincidences. *Proceedings of the National Academy of Sciences*, 107(52):22436–22441, 2010.

[16] D. J. Crandall, L. Backstrom, D. Huttenlocher, and J. Kleinberg. Mapping the world's photos. In *Proceedings of the 18th international conference on World wide web*, pages 761–770. ACM, 2009.

[17] K. De Bock and D. Van den Poel. Predicting website audience demographics forweb advertising targeting using multi-website clickstream data. *Fundamenta Informaticae*, 98(1):49–70, 2010.

[18] C. Ding, T. Li, W. Peng, and H. Park. Orthogonal nonnegative matrix tri-factorizations for clustering. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 126–135. ACM, 2006.

[19] N. Eagle, A. S. Pentland, and D. Lazer. Inferring friendship network structure by using mobile phone data. *Proceedings of the National Academy of Sciences*, 106(36):15274–15278, 2009.

[20] D. Easley and J. Kleinberg. *Networks, crowds, and markets: Reasoning about a highly connected world*. Cambridge Univ Pr, 2010.

[21] L. Fang and K. LeFevre. Privacy wizards for social networking sites. In *the 19th International World Wide Web Conference (WWW)*, 2010.

[22] L. A. Fast and D. C. Funder. Personality as manifest in word use: correlations with self-report, acquaintance report, and behavior. *Journal of personality and social psychology*, 94(2):334, 2008.

[23] B. Fung, K. Wang, R. Chen, and P. S. Yu. Privacy-preserving data publishing: A survey of recent developments. *ACM Computing Surveys (CSUR)*, 42(4):14, 2010.

[24] M. Garey and D. Johnson. *Computers and Intractability: A Guide to the Theory of NP-completeness*. WH Freeman & Co., 1979.

[25] E. Gilbert and K. Karahalios. Predicting tie strength with social media. In *the 27th ACM CHI*, pages 211–220, 2009.

[26] S. Goel, J. M. Hofman, and M. I. Sirer. Who does What on the Web: A Large-scale Study of Browsing Behavior. In *Proceedings of ICWSM*, 2012.

[27] J. Golbeck and D. Hansen. Computing political preference among twitter followers. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1105–1108. ACM, 2011.

[28] J. Golbeck, C. Robles, M. Edmondson, and K. Turner. Predicting personality from twitter. In *Privacy, security, risk and trust (passat), 2011 ieee third international conference on and 2011 ieee third international conference on social computing (socialcom)*, pages 149–156. IEEE, 2011.

[29] J. Golbeck, C. Robles, and K. Turner. Predicting personality with social media. In *CHI'11 Extended Abstracts on Human Factors in Computing Systems*, pages 253–262. ACM, 2011.

[30] S. D. Gosling, S. J. Ko, T. Mannarelli, and M. E. Morris. A room with a cue: personality judgments based on offices and bedrooms. *Journal of personality and social psychology*, 82(3):379, 2002.

[31] R. Gross and A. Acquisti. Information revelation and privacy in online social networks. In *the ACM workshop on Privacy in the electronic society*, pages 71–80. ACM, 2005.

[32] P. Gundecha, G. Barbier, and H. Liu. Exploiting Vulnerability to Secure User Privacy on a Social Networking Site. In *Proceedings of the 17th ACM SIGKDD*, 2011.

[33] P. Gundecha, G. Barbier, J. Tang, and H. Liu. User vulnerability and its reduction on a social networking site. *ACM Trans. Knowl. Discov. Data*, 9(2):12:1–12:25, Sept. 2014.

[34] P. Gundecha, S. Ranganath, Z. Feng, and H. Liu. A Tool for Collecting Provenance Data in Social Media. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1462–1465. ACM, 2013.

[35] B. Hecht, L. Hong, B. Suh, and E. H. Chi. Tweets from justin bieber's heart: the dynamics of the location field in user profiles. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 237–246. ACM, 2011.

[36] A. Ho, A. Maiga, and E. Aimeur. Privacy protection issues in social networking sites. In *IEEE AICCSA*, 2009.

[37] J. Hu, H.-J. Zeng, H. Li, C. Niu, and Z. Chen. Demographic Prediction based on User's Browsing Behavior. In *Proceedings of WWW*, 2007.

[38] X. Hu, J. Tang, H. Gao, and H. Liu. Unsupervised sentiment analysis with emotional signals. In *Proceedings of the 22nd international conference on World Wide Web*, pages 607–618. International World Wide Web Conferences Steering Committee, 2013.

[39] C. Jernigan and B. F. Mistree. Gaydar: Facebook friendships expose sexual orientation. *First Monday*, 14(10), 2009.

[40] R. Jones, R. Kumar, B. Pang, and A. Tomkins. I know what you did last summer: query logs and user privacy. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 909–914. ACM, 2007.

[41] M. Kosinski, D. Stillwell, and T. Graepel. Private Traits and Attributes are Predictable from Digital Records of Human Behavior. *Proceedings of the National Academy of Sciences*, 2013.

[42] B. Krishnamurthy and C. Wills. On the leakage of personally identifiable information via online social networks. *ACM SIGCOMM Computer Communication Review*, 40(1):112–117, 2010.

[43] T. Kucukyilmaz, B. B. Cambazoglu, C. Aykanat, and F. Can. Chat Mining for Gender Prediction. In *Advances in Information Systems*, pages 274–283. Springer, 2006.

[44] R. Li, S. Wang, H. Deng, R. Wang, and K. C.-C. Chang. Towards social user profiling: unified and discriminative influence model for inferring home locations. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1023–1031. ACM, 2012.

[45] T. Li, Y. Zhang, and V. Sindhwani. A Non-negative Matrix Tri-factorization approach to Sentiment Classification with Lexical Prior Knowledge. In *Proceedings of the ACL*, 2009.

[46] H. Liu and P. Maes. Interestmap: Harvesting social network profiles for recommendations. *Beyond Personalization*, 2005.

[47] W. Liu and D. Ruths. Whats in a name? using first names as features for gender inference in twitter. In *Analyzing Microtext: 2013 AAAI Spring Symposium*, 2013.

[48] G. Loewenstein, L. Thompson, and M. Bazerman. Social Utility and Decision Making in Interpersonal Contexts. *Journal of Personality and Social Psychology*, 57(3):426, 1989.

[49] Y. Lu, P. Tsaparas, A. Ntoulas, and L. Polanyi. Exploiting Social Context for Review Quality Prediction. In *Proceedings of WWW*, 2010.

[50] H. Ma, D. Zhou, C. Liu, M. R. Lyu, and I. King. Recommender systems with social regularization. In *Proceedings of WSDM*, 2011.

[51] B. Marcus, F. Machilek, and A. Schutz. Personality in Cyberspace: Personal Web Sites as Media for Personality Expressions and Impressions. *Journal of Personality and Social Psychology*, 90(6):1014–1031, 2006.

[52] M. McPherson, L. Smith-Lovin, and J. M. Cook. Birds of a Feather: Homophily in Social Networks. *Annual review of sociology*, pages 415–444, 2001.

[53] A. Mislove, B. Viswanath, K. P. Gummadi, and P. Druschel. You are who you know: inferring user profiles in online social networks. In *Proceedings of the third ACM international conference on Web search and data mining*, pages 251–260. ACM, 2010.

[54] D. Murray and K. Durrell. Inferring demographic attributes of anonymous internet users. In *Web Usage Analysis and User Profiling*, pages 7–20. Springer, 2000.

[55] A. Narayanan and V. Shmatikov. Robust de-anonymization of large sparse datasets. In *the 29th IEEE Symposium on Security and Privacy*, 2008.

[56] A. Narayanan and V. Shmatikov. De-anonymizing social networks. In *the 30th IEEE Symposium on Security and Privacy*, 2009.

[57] G. Nemhauser, L. Wolsey, and M. Fisher. An Analysis of Approximations for Maximizing Submodular Set Functions. *Mathematical Programming*, 14(1):265–294, 1978.

[58] T. Parsons. Social interaction. *International Encyclopedia of the Social Sciences*, 7:429–441, 1968.

[59] M. Pennacchiotti and A.-M. Popescu. A machine learning approach to twitter user classification. In *ICWSM*, 2011.

[60] D. Quercia, M. Kosinski, D. Stillwell, and J. Crowcroft. Our Twitter Profiles, Our Selves: Predicting Personality with Twitter. In *Proceedings of PASSAT and SOCIALCOM*, 2011.

[61] D. Rao, M. J. Paul, C. Fink, D. Yarowsky, T. Oates, and G. Coppersmith. Hierarchical bayesian models for latent attribute detection in social media. In *ICWSM*, 2011.

[62] D. Rao and D. Yarowsky. Detecting latent user properties in social media. In *Proc. of the NIPS MLSN Workshop*, 2010.

[63] D. Rao, D. Yarowsky, A. Shreevats, and M. Gupta. Classifying latent user attributes in twitter. In *Proceedings of the 2nd international workshop on Search and mining user-generated contents*, pages 37–44. ACM, 2010.

[64] P. J. Rentfrow and S. D. Gosling. The Do Re Mi's of Everyday Life: The Structure and Personality Correlates of Music Preferences. *Journal of personality and social psychology*, 84(6):1236–1256, 2003.

[65] R. Schifanella, A. Barrat, C. Cattuto, B. Markines, and F. Menczer. Folks in folksonomies: social link prediction from shared metadata. In *Proceedings of the third ACM international conference on Web search and data mining*, pages 271–280. ACM, 2010.

[66] D. Seung and L. Lee. Algorithms for Non-negative Matrix Factorization. In *Proceedings of the NIPS*, 2001.

[67] R. Z. Shamanth Kumar and H. Liu. Understanding User Migration Patterns Across Social Media. In *the 25th International Conference on Artificial Intelligence (AAAI)*, 2011.

[68] C. Song, Z. Qu, N. Blumm, and A.-L. Barabási. Limits of predictability in human mobility. *Science*, 327(5968):1018–1021, 2010.

[69] A. Squicciarini, M. Shehab, and F. Paci. Collective privacy management in social networks. In *the 18th international conference on World wide web (WWW)*, 2009.

[70] M. Sviridenko. A Note on Maximizing a Submodular Set Function Subject to a Knapsack Constraint. *Operations Research Letters*, 32(1):41–43, 2004.

[71] J. Tang, H. Gao, X. Hu, and H. Liu. Exploiting homophily effect for trust prediction. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 53–62. ACM, 2013.

[72] J. Tang, X. Hu, H. Gao, and H. Liu. Exploiting Local and Global Social Context for Recommendation. In *Proceedings of IJCAI*, 2013.

[73] J. Tang and H. Liu. Feature Selection with Linked Data in Social Media. In *SDM*, 2012.

[74] L. Tang and H. Liu. Relational learning via latent social dimensions. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 817–826. ACM, 2009.

[75] V. Vazirani. *Approximation Algorithms*. Springer, 2001.

[76] T. Veblen and M. Banta. *The Theory of the Leisure Class*. Oxford University Press, USA, 2007.

[77] G. Wang, T. Konolige, C. Wilson, X. Wang, H. Zheng, and B. Y. Zhao. You are how you click: Clickstream analysis for sybil detection. In *USENIX Security Symposium (Washington, DC, 2013)*, 2013.

[78] D. Watts and S. Strogatz. Collective Dynamics of Small-world Networks. *Nature*, 393(6684):440–442, 1998.

[79] G. Wondracek, T. Holz, E. Kirda, and C. Kruegel. A practical attack to de-anonymize social network users. In *the 31st IEEE Symposium on Security and Privacy*, 2010.

[80] F. Wong, C. W. Tan, S. Sen, and M. Chiang. Media, pundits and the us presidential election: Quantifying political leanings from tweets. In *Proceedings of the International Conference on Weblogs and Social Media*, 2013.

[81] R. Zafarani and H. Liu. Connecting users across social media sites: a behavioral-modeling approach. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 41–49. ACM, 2013.

[82] E. Zheleva and L. Getoor. To join or not to join: the illusion of privacy in social networks with mixed public and private user profiles. In *the 18th International World Wide Web Conference (WWW)*, 2009.

[83] V. W. Zheng, Y. Zheng, X. Xie, and Q. Yang. Collaborative location and activity recommendations with gps history data. In *Proceedings of the 19th international conference on World wide web*, pages 1029–1038. ACM, 2010.

[84] Y. Zheng, L. Zhang, X. Xie, and W.-Y. Ma. Mining interesting locations and travel sequences from gps trajectories. In *Proceedings of the 18th international conference on World wide web*, pages 791–800. ACM, 2009.

BIOGRAPHICAL SKETCH

Pritam Gundecha is a Ph.D. candidate of Computer Science and Engineering at Arizona State University. He obtained his B.E. and M.S. from Pune Institute of Computer Technology (PICT) in 2005 and Arizona State University (ASU) in 2010, respectively. He was awarded the 2014 ASU President's Award for Innovation, and various Student Travel Awards and Scholarships. His research interests include modeling different types of social media data using data mining and machine learning techniques. He has published innovative works in highly ranked journals and top conference proceedings such as TKDD, INFORMS, SIGKDD, CIKM and ASONAM. For his user vulnerability research on social media, He was interviewed by Huffington Post, ReadWriteWeb, Newscientist, and Toronto Star, and also reported by few dozens of other media sites. He has interned at IBM's T.J. Watson Research Center in 2014. Updated information can be found at `http://www.public.asu.edu/~pgundech/`.