Cross-Platform Identification of Anonymous Identical Users in Multiple Social Media Networks

Xiaoping Zhou, Xun Liang, *Senior Member, IEEE*, Haiyan Zhang, and Yuefeng Ma

Abstract—The last few years have witnessed the emergence and evolution of a vibrant research stream on a large variety of online social media network (SMN) platforms. Recognizing anonymous, yet identical users among multiple SMNs is still an intractable problem. Clearly, cross-platform exploration may help solve many problems in social computing in both theory and applications. Since public profiles can be duplicated and easily impersonated by users with different purposes, most current user identification resolutions, which mainly focus on text mining of users' public profiles, are fragile. Some studies have attempted to match users based on the location and timing of user content as well as writing style. However, the locations are sparse in the majority of SMNs, and writing style is difficult to discern from the short sentences of leading SMNs such as Sina Microblog and Twitter. Moreover, since online SMNs are quite symmetric, existing user identification schemes based on network structure are not effective. The real-world friend cycle is highly individual and virtually no two users share a congruent friend cycle. Therefore, it is more accurate to use a friendship structure to analyze cross-platform SMNs. Since identical users tend to set up partial similar friendship structures in different SMNs, we proposed the Friend Relationship-Based User Identification (FRUI) algorithm. FRUI calculates a match degree for all candidate User Matched Pairs (UMPs), and only UMPs with top ranks are considered as identical users. We also developed two propositions to improve the efficiency of the algorithm. Results of extensive experiments demonstrate that FRUI performs much better than current network structure-based algorithms.

Index Terms—Cross-platform, social media network, anonymous identical users, friend relationship, user identification

1 Introduction

In the last decade, many types of social networking sites have emerged and contributed immensely to large volumes of real-world data on social behaviors. Twitter,[1] the largest microblog service, has more than 600 million users and produces upwards of 340 million tweets per day [1]. Sina Microblog,[2] the primary Twitter-style Chinese microblog website, has more than 500 million accounts and generates well over 100 million tweets per day [2].

Due to this diversity of online social media networks (SMNs), people tend to use different SMNs for different purposes. For instance, RenRen,[3] a Facebook-style but autonymous SMN, is used in China for blogs, while Sina Microblog is used to share statuses (Fig. 1). In other words, every existent SMN satisfies some user needs. In terms of SMN management, matching anonymous users across different SMN platforms can provide integrated details on each user and inform corresponding regulations, such as targeting services provisions. In theory, the cross-platform explorations allow a bird's-eye view of SMN user behaviors. However, nearly all recent SMN-based studies focus on a single SMN platform, yielding incomplete data. Therefore, this study investigates the strategy of crossing multiple SMN platforms to paint a comprehensive picture of these behaviors.

Nonetheless, cross-platform research faces numerous challenges. As shown in Fig. 1, with the growth of SMN platforms on the Internet, the cross-platform approach has merged various SMN platforms to create richer raw data and more complete SMNs for social computing tasks. SMN users form the natural bridges for these SMN platforms. The primary topic for cross-platform SMN research is user identification for different SMNs. Exploration of this topic lays a foundation for further cross-platform SMN research.

User identification is also called user recognition, user identity resolution, user matching, and anchor linking. Although no solution can identify all identical anonymous SMN users, some SMN elements may be used to identify a portion of users across multiple SMNs. Many studies have addressed the user identification problem by examining public user profile attributes, including screen name, birthday, location, gender, profile photo, etc. [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17]. Since these attributes do not require exclusivity and are easily faked by users for different purposes (including malicious users), these schemes are quite fragile. Some researchers have leveraged public user activities to recognize users using post time, location and writing style [18], [19], [20], [21]. Since location data is difficult to obtain and writing style is difficult to extract from short sentences, these techniques are plagued by limitations. Although connections can be collected and are difficult to impersonate in nearly all SMNs, our literature review revealed only a few studies that explored employing user

1. http://www.twitter.com
2. http://www.weibo.com
3. http://www.renren.com

- *X. Zhou is with the Department of Computer Science, Renmin University of China, Beijing 100872, China, and the Beijing University of Civil Engineering and Architecture, Beijing 100044, China. E-mail: lukefchou@gmail.com.*
- *X. Liang, H. Zhang, and Y. Ma are with the Department of Computer Science, Renmin University of China, Beijing 100872, China. E-mail: {xliang, zhy_rabbit}@ruc.edu.cn, rzmyf1976@163.com.*

Manuscript received 30 Jan. 2015; revised 30 July 2015; accepted 21 Sept. 2015. Date of publication 1 Oct. 2015; date of current version 6 Jan. 2016. Recommended for acceptance by F. Bonchi
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TKDE.2015.2485222

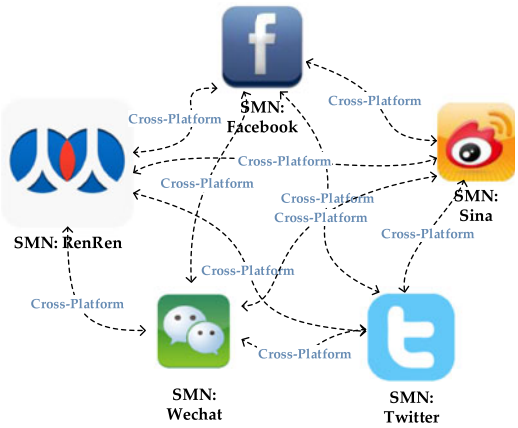


Fig. 1. Cross-platform research to merge a variety of SMNs.

friends to identify users [22], [23], [24]. In terms of data security and privacy, Narayanan and Shmatikov (NS for short) [22] de-anonymized a social network graph by correlating it with known identities. NS was the first effort to recognize users purely by using connections, and successfully matched 30 percent of the accounts with a 12 percent error rate. Bartunov et al. [23] proposed a Joint Link-attribute Algorithm (JLA) to match two social networks and obtained a portion of identical users. Korula and Lattanzi [24] utilized the degrees of unmapped users, as well as the number of common neighbors, to reconcile SMNs.

SMN connections fall into two categories: single-following connections and mutual-following connections. Single-following connections are also called following relationships or following links. If user A follows user B, then user A and user B have a following relationship (single-way fans in which one knows the other, but not vice versa). Following relationships are common in microblogging SMNs, such as Twitter and Sina Microblog. Likewise, mutual-following connections are called friend relationships. In microblogging SMNs, a friend relationship refers to the mutual following relationships between two users. In most other SMNs, such as Facebook, RenRen and Wechat, a friend relationship forms only if a friend request is sent by one user and confirmed by the other user. Friend relationships are difficult to fake by malicious users, and therefore reflect real-world relationships much better. Due to their reliability and consistency, friend relationships are more robust in user identification tasks. Moreover, since unified friend relationships are formed, our algorithm can also be applied to SMNs with a heterogeneous network structure, such as Twitter and Facebook.

In this study, we focused on friend relationships in SMNs and developed a new algorithm based on network structures. This algorithm can only identify a portion of the identical users in real-world SMNs. However, it can be applied jointly with other feature-based user identification algorithms for more accurate identification results. This study makes the following contributions to the research field by:

1) Developing a uniform solution framework for network structure-based user identification. First, a set of seed or priori mapped users are provided manually or otherwise identified. Iteration is used to re-identify as many users as possible, using the seed or priori mapped users along with network structures. In the current literature, all network structure-based solutions perform in this manner.
2) Proposing a novel Friend Relationship-based User Identification (FRUI) algorithm. In our analysis of cross-platform SMNs, we deeply mined friend relationships and network structures. In the real world, people tend to have mostly the same friends in different SMNs, or the friend cycle is highly individual. The more matches in two unmapped users' known friends, the higher the probability that they belong to the same individual in the real world. Based on this fact, we proposed the FRUI algorithm. Since FRUI employs a unified friend relationship, it is apt to identify users from a heterogeneous network structure. Unlike existing algorithms [22], [23], [24], FRUI chooses candidate matching pairs from currently known identical users rather than unmapped ones. This operation reduces computational complexity, since only a very small portion of unmapped users are involved in each iteration. Moreover, since only mapped users are exploited, our solution is scalable and can be easily extended to online user identification applications. In contrast with current algorithms [22], [23], [24], FRUI requires no control parameters.
3) Providing concrete demonstrations of FRUI performance with three synthetic networks and two major online SMNs in China: Sina Microblog and RenRen. The synthetic networks include Erdős-Rényi (ER) [25] random networks, Watts-Strogatz (WS) [26] small-world networks and Barabási-Albert preferential attachment model (BA) [27] networks. Findings show that FRUI is superior to NS in these networks. Moreover, FRUI is effective for the de-anonymization task, since the user identification task is similar to the de-anonymization problem.

This article proceeds as follows. Section 2 reviews related work on cross-platform user identification. Section 3 systematically presents terminology on user identification across SMN platforms, summarizes a uniform solution framework in network structure-based user identification solutions, and formally presents the problem definition in friend relationship-based networks. Section 4 discusses the methods to obtain Priori UMPs. Section 5 proposes the FRUI algorithm and describes techniques to reduce time complexity. Section 6 covers the experimental studies. Section 7 offers conclusions.

2 Related Work on Cross-Platform User Identification

Profiles, contents and network structures are three cardinal components in an SMN. Accordingly, current studies on cross-platform user identification can be divided into three categories: profile-based, content-based and network structure-based approaches.

2.1 Profile-Based User Identification

Several studies addressing anonymous user identification have focused on public profile attributes, including screen name, gender, birthday, city and profile image.

A screen name is the publically required profile feature in almost all SMNs. It has been widely explored as a way to recognize users across different SMNs. Perito et al. [3] calculated the similarity of screen names and identified users using binary classifiers. Similarly, Liu et al. [4] matched users in an unsupervised approach using screen names. Zafarani and Liu [5] proposed a method to map identities across different SMN platforms, empirically validating several hypotheses. On top of this work, they [6] further developed a user mapping method by modeling user behavior on screen names. Among public profile attributes, the profile image is another feature that has received considerable study. Acquisti et al. [7] addressed the user identification task with a face recognition algorithm. Although both screen name and profile image can identify users, they cannot be applied to large SMNs. This is because some users may have the same screen name and profile images. For example, many users have the screen name "John Smith" on Facebook.

Obviously, leveraging a combination of profile features can result in better user identification. Iofciu et al. [8] proposed an approach by measuring the distance between user profiles. Motoyama and Varghese [9] gathered attributes (education, occupation, etc.) as sets of words and matched users by calculating the similarity of users. Goga [10] linked accounts belonging to the same person identity, based solely on the profile information. Cortis [11] proposed a weighted ontology-based user profile resolution technique. Abel et al. [12] aggregated user profiles and matched users across systems. Similar studies across multiple platforms are also found in [13], [14], [15].

Undoubtedly, public profile attributes provide powerful information for user identification. However, some attributes are duplicated in large-scale SMNs, and are easily impersonated. Thus, purely profile-based schemes have limitations when they are applied to large-scale SMNs.

2.2 Content-Based User Identification

Content-Based User Identification solutions attempt to recognize users based on the times and locations that users post content, as well as the writing style of the content.

Zheng et al. [18] proposed a framework for authorship identification using the writing style of online messages and classification techniques. Almishari and Tsudik [19] proposed linking users across different SMNs by exploiting the writing style of authors. Kong and Zhang [20] proposed Multi-Network Anchoring (MNA) to map users. They calculated the combined similarities of user's social, spatial, temporal and text information in different SMNs, and examined a stable matching problem between two sets of user accounts. Goga et al. [21] exploited the geo-location attached to users' posts, the timestamp of posts, and users' writing style to address user identification tasks.

Geo-location appears to have forceful features for user recognition. However, this information is often sparse in SMNs, since only a small portion of users are willing to post their locations. Although writing style solutions perform well in scenarios involving long content, these techniques are not applicable to SMNs such as Twitter and Sina Microblog, in which short sentences are most likely posted.

2.3 Network Structure-Based User Identification

Network structure-based studies [22], [23], [24] on user identification across multiple SMNs are used to recognize identical users solely by user network structures and seed, or priori, identified users. As shown above, network-based user identification poses several major challenges, with few studies to build on [22], [23], [24].

To address this problem, Bartunov et al. [23] proposed an approach based on conditional random fields called Joint Link-Attribute (JLA). JLA considered both profile attributes and network properties. To analyze privacy and anonymity, Narayanan and Shmatikov [22] developed NS, based solely on network topology. Similar to FRUI, NS and JLA are one-to-one maps. To reconcile the SMNs, Korula and Lattanzi [24] presented a many-to-many mapping algorithm based on the degrees of unmapped users and the number of common neighbors, using two control parameters to fine-tune performance. These works had similar workflow, finding seed users first, then using these seed users to recursively propagate information through networks and extend sets of mapped nodes.

The task on user identification is closely related to the de-anonymization problem [28] for privacy-preserving social network analysis, which re-identifies individuals in online published SMN datasets. In this context, SMN data are anonymized before release. Zhou and Pei analyzed the neighborhood attacks of de-anonymization and proposed privacy preservation approaches using k-anonymity and l-diversity [29], [30]. Other de-anonymization attacks have also been analyzed [31], [32], [33]. Since cross-platform user identification is similar to the de-anonymization task, it can be applied to address the de-anonymization problem. As demonstrated in the experiments in Section 5, FRUI performs much better than NS, the de-anonymization algorithm.

The joint use of profile information, user behaviors hidden content and network structures may lead to better results. Jain and Kumaraguru [16], [17] developed Finding Nemo, a method that matches Facebook and Twitter accounts. However, this text-based network search method has low accuracy and high complexity in terms of user identification, since only the texts of the same nicknames are recognized when searching the friend sets of friends [12], [13]. Bartunov et al. [23] integrated profiles with a network structure using a Conditional Random Fields model and obtained better user identification results.

Network structure-based user identification is a hard nut to crack, and can be used to identify only a portion of identical users. NS, the first network structure-based user recognition algorithm across SMNs, can carry out user recognition tasks by using only the network structure, and identified 30.8 percent identical users in a ground-truth dataset [22]. Suppose that there are two SMNs: SMN_A and SMN_B. NS first calculates a set of mapping scores for each single, unmapped user entity (UE) in SMN_A to every unmapped user entity in SMN_B. Then an eccentricity is applied to determine whether or not a user in SMN_B can be matched. Only if the eccentricity is larger than a threshold would a user match be accepted. In addition, NS requires a *reverse match* to confirm the user match, which is costly in experiments.

JLA attempts to match unmapped nodes from different graphs by comparing the mapped neighbors of each

TABLE 1
Comparisons of FRUI to JLA and NS

Differences	JLA	NS	FRUI
Network Type	Undirected	Directed	Undirected
Additional Control Parameters	-	Eccentricity threshold	-
Matching Method	Unmapped neighbors of nodes from single graph	Unmapped users from different networks	Mapped users
Attributes Used	Identified users and their degrees	Identified users and in/out-degree of the unmapped users	Identified users
Match Degree	Dice coefficient	Shared known outgoing/incoming neighbors and in/out degree	Shared identified friends

node. It calculates a network distance between any two unmapped nodes in two different undirected networks. Similarly, empirical studies show that certain profiles can be matched based solely on the network structure using JLA.

In this study, we propose an innovative approach to address the challenges faced by previous studies. This new approach focuses on the friendship structure, and develops the Friend Relationship-based User Identification algorithm. FRUI differs from the two existing algorithms, JLA and NS, in the following aspects (see Table 1):

1) NS is suitable for directed networks, while JLA and FRUI focus on undirected networks. JLA is restricted in undirected networks by Conditional Random Fields, while FRUI relies on friend relationships, as this is more reliable and consistent with real-life friendship.
2) NS requires an additional control parameter (eccentricity threshold) to identify user matches. If the eccentricity is above a pre-determined threshold, NS accepts a candidate User Matched Pair (UMP). Clearly, the threshold is a free parameter and should be provided in advance. In contrast, no extra free parameters are required by JLA and FRUI.
3) JLA compares unmapped neighbors of nodes from one of the two SMNs, while NS matches unidentified users from different networks by comparing the mapped neighbors of each node. FRUI aims to identify the most matched pairs among mapped users, but does not iterate unmapped users. Therefore, it markedly reduces computational complexity.
4) NS employs unmapped users' in- and out-degrees, as well as the identified users, to calculate scores in directed networks. JLA, in contrast, employs identified users and their degrees. Any mapped user has different degrees in different SMNs. Therefore, details on how these degrees are obtained should be discussed in advance. Comparatively, only identified users are required in our FRUI.
5) NS computes the match degree by shared known outgoing/incoming neighbors and out-/in-degrees, with the assumption that users in different SMNs have similar outgoing/incoming neighbors. JLA uses a dice coefficient to calculate the match degree, which may mismatch users with high probability when two users share only a few known friends. In contrast, FRUI takes into account the number of shared known friends in match degree calculation. Moreover, the calculation method to match degree in FRUI has been shown to be simpler and more effective than those of JLA and NS (see Section 4.2).

3 Problem Definition in Cross-Platform User Identification

This section defines related terminologies, presents a uniform solution framework for user identification solely by using network structure, and defines the problem of friend relationship-based user mapping.

3.1 Terminology

Social media refers to virtual communities and networks in which people create, share, and/or exchange information and ideas [34]. In social media, people are allowed to (1) construct public or semi-public profiles within a bounded system, and (2) articulate with a set of other users with whom they share connections [35]. From this description, it is evident that an SMN is composed of three crucial elements: users with public or semi-public profiles, interaction information among users (or content), and connections (or network). Below are formal definitions of these terms.

Definition 1 (SMN). An SMN is defined as SMN = {U, C, I}, where U, C and I denote the users, connections and interactions among users, respectively.

Delving into the main components of an SMN, one can easily find that both C and I are generated by U. To this extent, C and I can be treated as the attributes of U, and U is the core item in the SMN. Therefore, user identification is of paramount importance in cross-SMN studies.

In this study, SMN_A is used to represent SMN A. Without a specification, SMN_A denotes the pure network structure of SMN A.

Definition 2 (User Entity). A User Entity is a user in combination with his or her profile, connections and interaction content. An SMN is a set of UEs which has the same number as the accounts in the SMN. Similarly, UE_A is used to indicate the UE list of SMN_A, and UE_{Ai} is taken as the token of the i-th element in UE_A.

Definition 3 (User matched pair). Given that SMN_A and SMN_B, if UE_{Ai} and UE_{Bj} belong to the same individual in real-life, which is denoted as Ψ, then we hold that UE_{Ai} and UE_{Bj} match on Ψ, and they compose a User Matched Pair UMP_Ψ. UMP_Ψ can also be expressed as $UMP_{A\sim B}(i, j)$ or $UMP(UE_{Ai}, UE_{Bj})$, equivalently.

Definition 4 (Fully overlapped SMNs). If a UMP set covers all users in both SMN_A and SMN_B, and the two SMNs have the same network structure, then all UEs and their

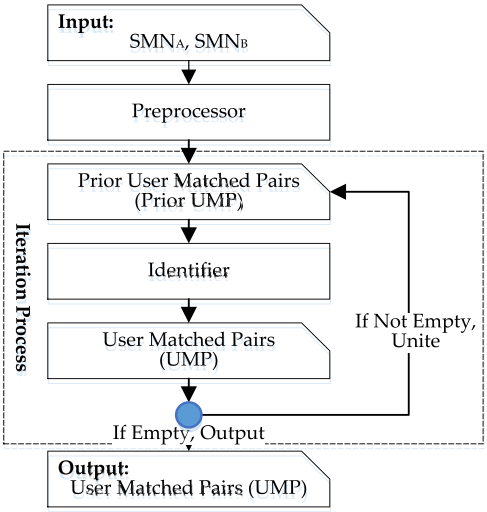


Fig. 2. Uniform solution framework. The network structure-based user identification first obtains Priori UMPs through a Preprocessor, and then identifies more UMPs through the Identifier in an iteration process.

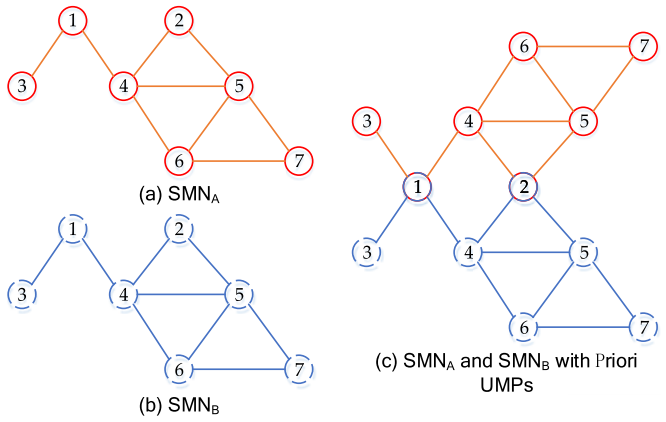


Fig. 3. Examples of two simple SMNs with FRUI. (a) and (b) are two simple SMNs, SMN$_A$ and SMN$_B$. (c) shows that SMN$_A$ and SMN$_B$ have two Priori UMPs 1 and 2, denoted as UMP$_{A\sim B}$(1, 1) and UMP$_{A\sim B}$(2, 2). Since user 4 in both SMNs shares the same and the largest friend set, UMP$_{A\sim B}$(4, 4) is identified and added to the Priori UMP set. Iteratively, UMP$_{A\sim B}$(5, 5), UMP$_{A\sim B}$(6, 6), UMP$_{A\sim B}$(7, 7), and UMP$_{A\sim B}$(1, 1) are identified consecutively.

relations overlap in the two SMNs and we can state that the SMN$_A$ and SMN$_B$ are fully matched.

Definition 5 (Priori UMP). Priori UMPs are UMPs given in advance, before user identification resolution work is executed. Priori UMPs are often used as the condition to identify more UMPs.

Definition 6 (Valid priori UMP). Valid Priori UMPs are Priori UMPs that are useful to user identification.

In FRUI, only the Priori UMPs connected to unmapped users in both SMNs are Valid Priori UMPs.

Definition 7 (Adjacent users). Unmapped users with connections to Priori UMPs are defined as Adjacent Users. Only Adjacent Users are involved in an iteration process in FRUI.

3.2 Uniform Solution Framework

According to the definitions above, identifying users across two SMNs yields a UMP set using Priori UMPs. Thus, network structure-based user identification algorithms are divided into two main steps: Priori UMP set recognition and iteration identification.

Fig. 2 shows the uniform solution framework. It has two modules: the Preprocessor and Identifier. The Preprocessor is set to reveal Priori UMPs through a limited number of profiles. However, the Identifier, the core component of our resolution, recognizes UMPs through users' networks in an iterative manner. In the uniform solution framework, the input is two SMNs in which user identification is performed, and the output is the UMPs. After a set of Priori UMPs is identified, the Identifier is implemented to recognize a set of new UMPs using network structure in the iteration process. The identified UMP set, if it is not empty, is in union with the Priori UMP set and yields to the new Priori UMPs for the next iteration. The iteration process ends when no UMP can be identified by the Identifier.

Different identification algorithms in the Identifier produce different results. Thus, most efforts in network structure-based user identification are placed on the Identifier, such as NS.

3.3 Problem Definition

In the real world, we can infer that each person has his own friend cycle, which is highly individual. Therefore, if we know all of a person's friends, we probably know who he is. Using SMN$_A$ in Fig. 3a as an example, if one has only user 1 as a friend, it is obvious that he must be user 3. If someone asks who has the friend set of users 1 and 2, it is obviously user 4. Users tend to have similar friends across different SMNs. Xuan and Wu's survey revealed that a user in QQ, one of the most famous Instant Messengers in China, shared telephone numbers of about 60 percent of his QQ friends [36]. We investigated 129 individuals with both RenRen and Sina Microblog accounts and found that an average of 67.5 percent of their friends in Sina Microblog concurred in RenRen. Numbers of their Sina Microblog friends varied from 4 to 317. Consequently, we can hypothesize that: (1) If some Valid Priori UMPs are given, then a set of candidate UMPs can be deduced, and (2) the more known friends are shared in a candidate UMP, the higher the probability that they belong to the same individual. Using the fully overlapped SMN$_A$ and SMN$_B$, as an illustration, Fig. 3c shows SMN$_A$ and SMN$_B$ with the Priori UMPs, UMP$_{A\sim B}$(1, 1) and UMP$_{A\sim B}$(2, 2). Intuitively, (UE$_{A1}$, UE$_{B1}$) and (UE$_{A2}$, UE$_{B2}$) are placed together. Then we find that UE$_{A4}$ and UE$_{B4}$ share the same friend set, which is the largest set based on the current UMP set. We can then conclude that UE$_{A4}$ and UE$_{B4}$ stand a good chance of forming a new UMP. After UMP$_{A\sim B}$(4, 4) is identified, it is added to Priori UMPs. By repeatedly using the above method, UMP$_{A\sim B}$(5, 5), UMP$_{A\sim B}$(6, 6), UMP$_{A\sim B}$(7, 7), and UMP$_{A\sim B}$(1, 1) are identified consecutively.

The main question in the above scenario is the overlap of the users' friends. To address this issue, we discuss the overlap of SMNs, including node and edge overlap, below.

1) Node overlap. Many studies have verified that numerous users are overlapped in different SMNs. Nearly all cross-platform user identification studies mention node overlap, because it is the fundamental assumption to solve this issue. Early in 2007,

64 percent of Facebook users had MySpace accounts [37]. Recently, Goga et al. [10] noted that many users have accounts on Google+, MySpace, Twitter, Facebook and Flickr. Since many users overlap in terms of MSNs, the question remaining is whether the relationships overlap as well.

2) Edge overlap. Until very recently, no statistical studies quantified relationship overlap in two SMNs. However, some studies noted that these relationships overlap to a certain extent. NS [22], which identifies users purely through networks in ground-truth datasets, proved that users have similar relationships in Twitter and Flickr. Paridhi [16], [17] also found that users tend to connect with a segment of the same people across SMNs, and introduced network structure to improve the accuracy of user identification between Twitter and Facebook. All these network structure involved user identification solutions conceal the fact that edges are partially overlapped in different SMNs. The reasons for edge overlap may be that: (1) many people tend to set up relations with their real-life friends (e.g., classmates, workmates, and family members) in different SMNs. (2) People tend to connect to those with similar interests. (3) In directed SMNs, users are allowed to be followed by their "fans."

As discussed in Section 1, the friend relationship is much more reliable than the unknown "fan" relationship, and nearly all the SMNs support friend relationship (mutual fans in Microblogging SMNs). In this study, we leveraged the friend relationship to explore a new method of identifying users.

If two users from two different SMNs share enough friends, they are highly likely to be the same individual. The more matched their friends are, the higher the chance that they are the same person. The user identification problem can be defined as

$$f(\mathrm{UE}_{\mathrm{A}i}, \mathrm{UE}_{\mathrm{B}j}) = g(M_{ij}) = \begin{cases} 1, \text{if } \mathrm{UE}_{\mathrm{A}i} \text{ and } \mathrm{UE}_{\mathrm{B}j} \text{ belong to the same individual;} \\ 0, \text{otherwise,} \end{cases} \quad (1)$$

where M_{ij} denotes the match degree of $\mathrm{UE}_{\mathrm{A}i}$ and $\mathrm{UE}_{\mathrm{B}j}$'s known (identified) friends. The user identification problem is converted into the calculation of M_{ij} and the definition of function g in FRUI.

For various reasons, some people hold many accounts in the same SMN, yet we often assume that these multiple accounts are independent and belong to different individuals. In other words, we only identify one of these accounts.

4 Preprocessor

A preprocessor is designed to acquire as many Priori UMPs as possible. Currently, there is no common approach available to obtain UMPs between two SMNs. Specified methods must be formulated according to given SMNs. Although no unified process is suitable for the Preprocessor, some algorithms can be adopted according to the application, e.g., email address, screen name, URL, etc.

An email address appears to be a unique feature for each account, and can be used to collect Priori UMPs. Balduzzi et al. [38] explored email addresses to find identical users among different SMNs with the "Friend Finder" mechanism. However, since email addresses are private, nearly all SMNs have disabled the "Friend Finder."

As stated in [17], one individual tends to use the same nickname in different SMN platforms. Thus, when a nickname can be taken as the UE, the nickname can be obtained and is unique. In these cases, the pair of UEs having the same nickname from two SMNs can be treated as a UMP. However, in scenarios where people are allowed to have the same or similar usernames such as RenRen, this method fails to identify users. One solution is to verify candidate UMPs through other accessible factors, such as description, location, and birthday.

With advances in SMN services, more SMNs allow users to bind their accounts with other major SMNs. In this case, priori knowledge can be obtained with bound information. For example, PaPa and ChangBa, two major mobile applications (apps) in China, encourage users to link their Sina Microblog accounts for commercial interests, bridging their websites with the largest microblog service in China.

As discussed in [16], Twitter provides an attribute, called a URL, for user self-identification. Preprocessors can directly use URLs to match a Twitter account to Facebook or other SMN accounts.

When no extra information except the network structure can be employed, the seed identification approach in NS [22] and the de-anonymization attacks in [28] are alternatives for the Preprocessor.

5 Identifier

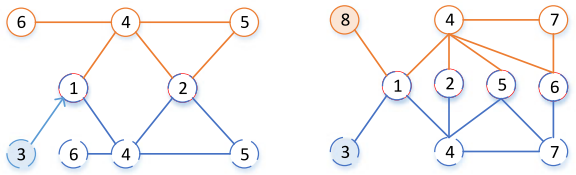
In this section, we systematically discuss our solution to the user identification problem by leveraging users' friends, and develop two propositions to improve the efficiency of our algorithm.

5.1 Methodology

The identifier finds UMPs using connections among users and Priori UMPs. As noted above, a match degree for each candidate UMP should be calculated in advance. NS formulates the match degree using in- and out-degrees in directed networks,

$$M_{ij} = s(\mathrm{UE}_{\mathrm{A}i}, \mathrm{UE}_{\mathrm{B}j}) = \frac{c_{\mathrm{in}}}{\sqrt{d_{\mathrm{in-B}j}}} + \frac{c_{\mathrm{out}}}{\sqrt{d_{\mathrm{out-B}j}}}, \quad (2)$$

where c_{in} and c_{out} denote the numbers of shared incoming and outgoing neighbors of $\mathrm{UE}_{\mathrm{A}i}$ and $\mathrm{UE}_{\mathrm{B}j}$ respectively, and $d_{\mathrm{in-B}j}$ and $d_{\mathrm{out-B}j}$ stand for the in- and out-degrees of $\mathrm{UE}_{\mathrm{B}j}$. NS operates under the assumption that the same user in different SMNs has the same amount of in- and out-degrees. In NS, M_{ij} depends heavily on $d_{\mathrm{in-B}j}$ and $d_{\mathrm{out\text{-}B}j}$. In single-following connections, users can follow any other users freely, which would introduce noise for the user identification task. We take Fig. 4a as an example, when $\mathrm{UE}_{\mathrm{B3}}$ follows $\mathrm{UE}_{\mathrm{B1}}$, $M_{43} = 1$ in NS. Once $\mathrm{UE}_{\mathrm{B4}}$ has a large in- or out-degree, NS has difficulty identifying $\mathrm{UMP}_{\mathrm{A}\sim\mathrm{B}}(4, 4)$. Nevertheless, our datasets (Fig. 10) and [39] indicate that real-world SMNs are symmetric, with many nodes sharing a portion of neighbors in SMNs. This would prevent identification of many identical users. Fig. 4b displays undirected graphs. Although $\mathrm{UE}_{\mathrm{B3}}$ and $\mathrm{UE}_{\mathrm{A4}}$ share only one known



(a) Noise in directed graphs (b) Noise in undirected graphs

Fig. 4. Noise in user identification. The nodes and links at the top, bottom and middle denote SMN_A, SMN_B and identified users, respectively. (a) shows the noise introduced in directed graphs, and a link with no arrow indicates a mutual-following relationship. A following relationship from UE_{B3} to identified user 1 would prevent identification of $UMP_{A\sim B}(4, 4)$, using NS when user 4 has a large in- or out-degree. This is the same as the friend relationship between users 3 and 1 in (b). In (b), the fact that M_{83} is larger than M_{44} using JLA results in the identification of an incorrect $UMP_{A\sim B}(8, 3)$.

friend, this may hinder identification of $UMP_{A\sim B}(4, 4)$. In other words, even though as many as 10 out of 100 friends are observed between UE_{A100} and UE_{B100}, any UE with a much lower degree that happens to share one identified user with UE_{A100} may hinder identification of $UMP_{A\sim B}(100, 100)$. Consequently, NS may miss numerous matched users, especially in the sparse SMNs.

As discussed above, the friend relationship requires confirmation by the two users, and is much more reliable and consistent in SMNs. Thus, it can reduce the noise introduced by a discretionary single-following relationship. Making use of the friend relationship in undirected networks, JLA defines the match degree as,

$$M_{ij} = s(\mathrm{UE}_{\mathrm{A}i}, \mathrm{UE}_{\mathrm{B}j}) = \frac{2 \times w(F_{Ai} \cap F_{\mathrm{B}j})}{w(F_{\mathrm{A}i}) + w(F_{\mathrm{B}j})}, \tag{3}$$

where $\cap$ represents the intersection operation of two sets, $\mathrm{F}_{\mathrm{A}i}$ and $\mathrm{F}_{\mathrm{B}j}$ denote the identified friend sets of $\mathrm{UE}_{\mathrm{A}i}$ and $\mathrm{UE}_{\mathrm{B}j}$, respectively. Generally, $w(F) = \sum_{v \in F} 1/d(v)$, where F represents a set of users, and $d(v)$ refers to the degree of user v. Apparently, M_{ij} returns 1 when $F_{Ai} = F_{Bj}$, and it often matches incorrectly when $|F_{Ai}|$ is not large enough. Taking $\mathrm{UE}_{\mathrm{A8}}$ and $\mathrm{UE}_{\mathrm{B3}}$ in Fig. 4b as an example, $\mathrm{F}_{\mathrm{A8}} = \mathrm{F}_{\mathrm{B3}} = \{\mathrm{UE}_{\mathrm{A1}}\}$, resulting in $M_{83} = 1$ using JLA. Consequently, JLA powerfully confirms that $\mathrm{UE}_{\mathrm{A8}}$ and $\mathrm{UE}_{\mathrm{B3}}$ are identical. However, $\mathrm{UE}_{\mathrm{A8}}$ and $\mathrm{UE}_{\mathrm{B3}}$ share only one known neighbor, which will likely lead to incorrect identification.

For any two SMNs, SMN_A and SMN_B can be considered as mirrors of the real world. Suppose that people set up random friendships in the real world; then the probability of a friendship between any two persons is p $(0 < p < 1)$, and for any friendship, s_a $(0 < s_a < 1)$ and s_b $(0 < s_b < 1)$ are probabilities that it exists in SMN_A and SMN_B, respectively. Therefore, the probabilities that a friendship exists in SMN_A and SMN_B are ps_a and ps_b, respectively. Note that a friendship exists in both SMN_A and SMN_B with the probability of ps_as_b. Subsequently, $\mathrm{UE}_{\mathrm{A}i}$ and $\mathrm{UE}_{\mathrm{B}j}$ share $|F|ps_as_b$ recognized friends when they belong to the same individual, where $|\mathrm{F}|$ denotes the number of recognized users. $\mathrm{UE}_{\mathrm{A}i}$ and $\mathrm{UE}_{\mathrm{B}j}$ share $|F|p^2s_as_b$ known friends when they are owned by different individuals. Since the difference in shared known friends accounts for $1 / p$ times between $\mathrm{UE}_{\mathrm{A}i} = UE_{Bj}$ and $UE_{Ai} \neq UE_{Bj}$, the match degree of $\mathrm{UE}_{\mathrm{A}i}$ and $\mathrm{UE}_{\mathrm{B}j}$ can be calculated as the number of shared known friends,

$$M_{ij} = |F_{\mathrm{A}j} \cap F_{\mathrm{B}j}|. \tag{4}$$

Apparently, s represents the overlap between the two SMNs, and p represents the density of the original SMN. For a given p, the larger s indicates the larger differences in the shared known friends between $\mathrm{UE}_{\mathrm{A}i} = \mathrm{UE}_{\mathrm{B}j}$ and $\mathrm{UE}_{\mathrm{A}i} \neq \mathrm{UE}_{\mathrm{B}j}$, and the smaller $|F|$ is required for the identification. For a given s, a lower p guarantees a larger gap in the shared known friends between $\mathrm{UE}_{\mathrm{A}i} = \mathrm{UE}_{\mathrm{B}j}$ and $\mathrm{UE}_{\mathrm{A}i} \neq \mathrm{UE}_{\mathrm{B}j}$. However, this requires a larger $|F|$ to ensure enough shared known friends between identical users. Conversely, when p is too large, the SMN turns out to be a clique. As a result, the nodes are too similar in network structure to be distinguished, and more Priori UMPs are required for the identification. Consequently, neither a larger nor a smaller p is propitious to the user identification tasks.

Although (4) makes up for the drawback of JLA in terms of match degree, it may generate Controversial UMPs. When $|F_{Ai} \cap F_{Bj}|$ and $|F_{Ak} \cap F_{Bj}|$ are equal and are sufficiently large, they have the same number of shared known friends. The question is which item is the counterpart of $\mathrm{UE}_{\mathrm{B}j}$, $\mathrm{UE}_{\mathrm{A}i}$ or $\mathrm{UE}_{\mathrm{A}k}$? $\mathrm{UMP}_{\mathrm{A}\sim\mathrm{B}}(i, j)$ and $\mathrm{UMP}_{\mathrm{A}\sim\mathrm{B}}(i, k)$ are called **Controversial UMPs**. Like NS, a Controversial UMP is created because real-world complex networks are symmetric. To solve this problem, we introduced the similarity of known friends to detail M_{ij}. Consequently, when no un-Controversial UMP exists, (4) is developed into

$$M_{ij} = \left|\mathrm{F}_{\mathrm{A}i} \cap \mathrm{F}_{\mathrm{B}j}\right| + \frac{\left|\mathrm{F}_{\mathrm{A}i} \cap \mathrm{F}_{\mathrm{B}j}\right|}{\min(|\mathrm{F}_{\mathrm{A}i}|, |\mathrm{F}_{\mathrm{B}j}|)}, \tag{5}$$

where $\min(|F_{Ai}|, |F_{Bj}|)$ takes the minimal value between $|F_{Ai}|$ and $|F_{Bj}|$. The higher value of M_{ij} hints at the matches between the two users, increasing the probability that it is a UMP.

According to the definition of M_{ij}, FRUI must formulate the function g. The more closely matched the known friends of a candidate UMP, the higher the chance that they belong to the same real identity. Hence, we formulate the following equation,

$$g(M_{ij}) = (M_{ij} == \max_{\mathrm{u}}(M)), \tag{6}$$

where M represents the match degrees of all candidate UMPs. In (6), $a == b$ returns 1 when a equals to b, and 0 otherwise. For any $u \in M$, denote $\Gamma(u)$ as the un-Controversial UMP set with a match degree of no less than u. We have

$$\max_{\mathrm{u}}(M) = \max(u), \mathrm{s.t} \quad \Gamma(u) \text{ is not empty}. \tag{7}$$

In practice, $\Gamma(u)$ is formed using (4). If no un-Controversial UMPs exist in$\Gamma(u)$, $\Gamma(u)$ will be recalculated using (5).

The question remains as to whether a control parameter is required to ensure a minimal bound of $\max_{\mathrm{u}}(M)$ in g. In this study, we did not set a minimal value for g. Because UMPs with higher match degrees are more likely to hit $\max_{\mathrm{u}}(M)$, $\max_{\mathrm{u}}(M)$ has difficulty dropping to those with fairly low match degrees. Although a threshold can be set to improve accuracy, no evaluation method provides the boundary in advance. Furthermore, the algorithm may stop too soon to identify more potential identical users. In short,

it is not necessary to set a limitation on g. Thus, our algorithm requires no additional control parameters, which is a major strength.

5.2 Algorithm

Clearly, the number of the shared known friends is the key value to be calculated in FRUI. To lower the complexity, we present the following two propositions.

Proposition 1. Given two SMNs, SMN_A and SMN_B, with s pairs of Priori UMPs, the $m \times n$ matrix $R = Q_A P_B$ contains the numbers of shared known friends, where m and n are the numbers of Adjacent Users in SMN_A and SMN_B, r_{ij} stands for the number of shared friends of UI_{Ai} and UI_{Bj} in the s pairs of UMP, Q_A and P_B denote the connections between Adjacent Users and identified users in SMN_A and the connections between identified users and Adjacent Users in SMN_B, respectively.

Proof. Q_A represents the connections between Adjacent Users and identified users, and can be written as $Q_A = [\alpha_1^T, \alpha_2^T, \ldots, \alpha_m^T]^T$ where $(\bullet)^T$ is the transposition of $(\bullet), \alpha_i \in \{0,1\}^{1\times s}$ denotes the connections of the ith Adjacent User to identified users. Similarly, P_B can be written as $P_B = [\beta_1, \beta_2, \ldots, \beta_n]$, where $\beta_j \in \{0,1\}^{s\times 1}$ denotes the connections of the j-th Adjacent User to identified users. As a result, matrix R can be converted to

$$R = Q_A P_B = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_m \end{bmatrix} [\,\beta_1 \quad \beta_2 \quad \cdots \quad \beta_n\,]. \tag{8}$$

Considering $\alpha_i\beta_j$ with $\alpha_i = [a_{i1}a_{i2}\ldots a_{is}]$ and $\beta_j = [b_{j1}b_{j2}\ldots b_{js}]^T$, where a_{ik} and b_{jk} denote whether there is a connection between the ith Adjacent User and the kth identified user in SMN_A, and the jth Adjacent User and the kth identified user in SMN_B, respectively. If both the ith Adjacent User in SMN_A and jth Adjacent User in SMN_B are connected with the kth identified user, $a_{ik}b_{jk}$ is assigned 1, and 0 otherwise,

$$a_{ik}b_{jk} = \begin{cases} 1, & \text{if } UE_{\mathrm{Ai}} \text{ and } UE_{\mathrm{B}j} \text{ share the } k\text{-th identified user;} \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

Then, $\alpha_i\beta_j$ can be calculated as

$$\alpha_i\beta_j = \sum_{k=1}^{s} a_{ik}b_{jk}. \tag{10}$$

Obviously, the ith Adjacent User in SMN_A and the j-th Adjacent User in SMN_B share $\alpha_i\beta_j$ identified users. □

Since only Adjacent Users are involved, this greatly reduces computational complexity. Proposition 1 reduces complexity to $O(\sum_s d_{Ai}d_{Bi}) \le O(sd_Ad_B)$ in calculating the numbers of shared known friends when the s pairs are calculated separately, where d_A and d_B denote the maximal degrees of users in SMN_A and SMN_B, respectively.

Consider that k UMPs are identified in the tth iteration. Taking these k Priori UMPs as the input for Proposition 1, we have another matrix $\Delta R = \Delta Q_A \Delta P_B$, where ΔR, ΔQ_A, and ΔP_B have a similar meaning as R, Q_A, and P_B in Proposition 1. The combination of $R^{(t)}$ and ΔR returns $R^{(t+1)}$, where $R^{(t)}$ denotes matrix R for the tth iteration. This leads to Proposition 2.

Proposition 2. In the tth iteration process, if k UMPs are generated, then

$$R^{(t+1)} = combine(R^{(t)}, \Delta R), \tag{11}$$

where the function *combine* removes the items with any UE included in the k UMPs, and returns the *union* of the remaining ones in $R^{(t)}$ and ΔR. The *union* operation adds the value of the items in both $R^{(t)}$ and ΔR, and joins the left items.

Proof. Since those items with any UE in the k UMPs will not be used in subsequent identifications, they are removed. For those candidate UMPs that occur in both $R^{(t)}$ and ΔR, their shared known friends are the sum of those in $R^{(t)}$ and ΔR. The shared known friends of the candidate UMPs only existing in $R^{(t)}$ are unchanged. The number of the shared known friends of the new candidate UMPs generated by the k UMPs is the one in ΔR. □

It is worth-noting that only users in both SMNs with connections to newly identified users are involved in each iteration process, based on Proposition 2. Furthermore, when $t = 0, R^{(t)}$ is a **0** matrix, the k identified UMPs turn to Priori UMPs, then, Proposition 2 is degenerated into Proposition 1.

In the implementation, the Identifier first calculates matrix R using Proposition 1 and initializes the match degree. Then it iterates and identifies UMPs using function g until no UMP can be identified. In each iteration, once the UMPs are identified, the items are removed from the Candidate UMP list, and R is recalculated based on Proposition 2. The process is summarized in Algorithm 1.

Suppose that there are s Valid Priori UMPs in any iteration. Lines 4-11 in Algorithm 1 remove the identified UMPs and update the maximum match degree, and the time complexity costs $O(s) + O(min(v_A, v_B)) = O(min(v_A, v_B))$, where v_A and v_B denote the numbers of the users in SMN_A and SMN_B, respectively. Lines 12-19 update the Candidate UMP list and the maximum match degree using Propositions 1 and 2. As discussed above, the computational complexity is $O(sd_Ad_B)$. Lines 20-29 identify identical users for the next iteration using (6). Normally, the $max_u(M)$ can be found in the candidate UMPs with the largest M_{ij}, and the time complexity is no more than $O(min(v_A, v_B))$. In summary, the complexity of FRUI is $O(min(v_A, v_B)) + O(sd_Ad_B) + O(min(v_A, v_B)) \le O(min(v_A, v_B)d_Ad_B)$. Obviously, the complexity of FRUI is lower than $O((e_A + e_B)d_Ad_B)$ in NS [19], where e_A and e_B are the numbers of the edges in the two SMNs.

6 Experimental Studies

To evaluate the identification resolution, we verify FRUI in both synthetic and ground-truth networks. All the experiments were conducted in the computer with 8 G memory and 2.8 GHz CPU.

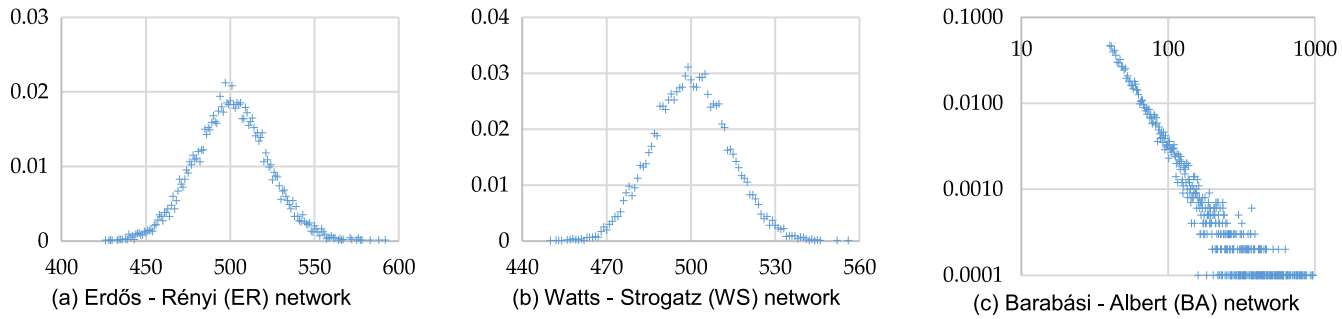


Fig. 5. Degree distribution of synthetic networks.

Algorithm 1. FRUI

Input: SMN_A, SMN_B, Priori UMPs: PUMPs
Output: Identified UMPs: UMPs

function FRUI(SMN_A, SMN_B, PUMPs)
  $T = \{\}, R = \text{dict}(), S = \text{PUMPs}, L = [], \max = 0, \text{F}_\text{A} = [], \text{F}_\text{B} = []$
  while S is not empty **do**
    Add S to T
    if $\max > 0$ **do**
      Remove S from L[max]
      while L[max] is empty
        $\max = \max - 1$
        if $\max == 0$ **do**
          return UMPs
        Remove UMPs with mapped UE from L[max]
    foreach $\text{UMP}_{\text{A}\sim\text{B}}(i, j)$ in S **do**
      foreach $\text{UE}_{\text{A}a}$ in the unmapped neighbors of $\text{UE}_{\text{A}i}$ **do**
        $\text{F}_\text{A}[i] = \text{F}_\text{A}[i] + 1$
        foreach $\text{UE}_{\text{A}b}$ in the unmapped neighbors of $\text{UE}_{\text{A}j}$ **do**
          $R[\text{UMP}_{\text{A}\sim\text{B}}(a, b)] += 1, \text{F}_\text{B}[j] = \text{F}_\text{B}[j] + 1$
          Add $\text{UMP}_{\text{A}\sim\text{B}}(a, b)$ to $L[R[\text{UMP}_{\text{A}\sim\text{B}}(a, b)]]$
          if $R[\text{UMP}_{\text{A}\sim\text{B}}(a, b)] > \max$ **do**
            $\max = R[\text{UMP}_{\text{A}\sim\text{B}}(a, b)]$
    $m = \max, S = \{\}$
    while S is empty **do**
      Remove UMPs with mapped UE from L[max]
      $C = L[m], m = m - 1, n = 0$
      $S =$ {un-Controversial UMPs in C}
      while S is empty **do**
        $n = n + 1, I =$ {UMPs with top n M_{ij} in C using (5)}
        $S =$ {un-Controversial UMPs in I}
        if $I == C$ **do**
          break
  return T

We used NS as a baseline because it is closest to FRUI as a state-of-art, network structure-based user identification algorithm, while JLA performs better when profile attributes are added. Without a loss of generality, the eccentricity threshold of NS is set to 0.5 in the experiments.

6.1 Synthetic Network Experiments

To validate the performance of FRUI, we conducted experiments in Erdős-Rényi [25] random networks, Watts-Strogatz [26] small-world networks and Barabási-Albert preferential attachment model (BA) [27] networks. Fig. 5 demonstrates the degree distribution of the three synthetic networks with 10,000 nodes. For the degree distribution, the ER, WS and BA networks followed a normal distribution, a bell distribution and a power-law distribution, respectively. The degree distribution of the WS network was similar to that of the ER network because both ER and WS networks are generated from the regular random network by rewiring each edge with a probability. If all edges are rewired so that the probability of rewiring equals 1, the network turns out to be an ER network; otherwise, it is a WS network. In the experiments, the probability of rewiring in WS network was 0.5.

We generated 10 pairs of networks in experiments to illustrate the performance of FRUI in synthetic networks. In the ER and WS network experiments, five networks with 5,000 nodes and another five with 10,000 nodes were created, with p equaling 0.05, 0.1, 0.2, 0.3 and 0.4, respectively. Similarly, in the BA network experiment, five networks with 10,000 nodes and another five with 20,000 nodes were produced. The number of edges to attach a new node to existing nodes, denoted as m, increased from 20 to 100 by 20. Subsequently, 30 pairs of networks were generated by 30 synthetic networks, with $s_a = s_b = 0.4$.

Table 2 displays results of empirical testing in ER networks. FRUI identified almost all identical nodes in the 10 pairs of networks, with only 2 percent UMPs. Table 3 illustrates the performance of FRUI in WS networks. Analogous to the experiments in the ER network, FRUI recognized no less than 75.9 percent of all UMPs, with 2 percent UMPs, and no less than 96.1 percent of all the UMPs with 5 percent UMPs. Table 4 shows that FRUI also has good performance in BA networks. No less than 89.4 percent of all UMPs can be identified by 5 percent UMPs. In all experiments on ER, WS and BA networks, FRUI revealed nearly all UMPs, with 5 percent UMPs. This indicates that FRUI can address the user identification task with a small portion of UMPs.

TABLE 2
Recall Rate of FRUI in ER Networks with $s_a = s_b = 0.4$

Nodes	Priori UMPs	$p = 0.05$	$p = 0.1$	$p = 0.2$	$p = 0.3$	$p = 0.4$
5000	0.01	0.985	0.998	0.994	0.017	0.004
	0.02	1	1	1	1	0.997
	0.03	1	1	1	1	1
	0.04	1	1	1	1	1
	0.05	1	1	1	1	1
10000	0.01	1	1	1	1	1
	0.02	1	1	1	1	1
	0.03	1	1	1	1	1
	0.04	1	1	1	1	1
	0.05	1	1	1	1	1

TABLE 3
Recall Rate of FRUI in WS Networks with $s_a = s_b = 0.4$

Nodes	Priori UMPs	$p = 0.05$	$p = 0.1$	$p = 0.2$	$p = 0.3$	$p = 0.4$
5,000	0.01	0.040	0.023	0.017	0.009	0.008
	0.02	0.759	0.928	0.993	0.991	0.993
	0.03	0.780	0.984	0.998	0.998	1
	0.04	0.964	0.994	0.999	1	1
	0.05	0.961	1	1	1	1
10,000	0.01	0.027	0.052	0.997	0.997	0.991
	0.02	0.899	0.997	1	1	1
	0.03	0.993	1	1	1	1
	0.04	0.998	1	1	1	1
	0.05	0.999	1	1	1	1

TABLE 4
Recall Rate of FRUI in BA Networks with $s_a = s_b = 0.4$

Nodes	Priori UMPs	$m = 20$	$m = 40$	$m = 60$	$m = 80$	$m = 100$
10,000	0.01	0.012	0.008	0.008	0.013	0.012
	0.02	0.637	0.980	0.983	0.977	0.962
	0.03	0.813	0.990	0.994	0.991	0.992
	0.04	0.882	0.996	0.998	0.998	0.996
	0.05	0.894	0.998	0.999	0.998	0.998
20,000	0.01	0.834	0.975	0.053	0.936	0.888
	0.02	0.889	0.994	0.998	0.997	0.996
	0.03	0.909	0.997	0.999	1	1
	0.04	0.916	0.998	1	1	1
	0.05	0.919	0.999	1	1	1

We also conducted experiments to compare the efficiency of FRUI and NS. Fig. 6 compares FRUI and NS in the three synthetic networks. Figs. 6a and 6b displays results of experiments conducted with $p = 0.05$ in networks generated by ER and WS models, with 1,000 and 5,000 nodes. Comparisons of FRUI and NS in ER and WS networks with 10,000 nodes were not illustrated, since both FRUI and NS identified almost all the UMPs, with 2 percent Priori UMPs in these networks. Fig. 6c displays empirical testing results in the BA network with $m = 20$. When the percent of Priori UMPs increased, the recall rates of both FRUI and NS increased, and FRUI identified more UMPs than NS in almost all scenarios in the three synthetic networks. Fig. 6d compares the precisions of FRUI and NS, with 1,000 nodes in ER and WS networks and 10,000 nodes in BA networks. It is obvious that FRUI outperforms NS. We also observed the performance of FRUI and NS in the networks with different densities. Fig. 7 presents the recall rates and precisions in the three synthetic networks with 5 percent Priori UMPs. Clearly, FRUI can identify more UMPs than NS with higher precision, especially in BA networks. Although the eccentricity threshold can be lowered to identify more UMPs, the precision will drop accordingly. In sum, these findings demonstrate that FRUI is more efficient than NS in identifying nodes in the three synthetic networks.

Figs. 7a and F7b indicates that when $p = 0.05$ and 0.4, the performance of both FRUI and NS decreased in the networks with 1,000 nodes. This is because extra Priori UMPs were necessary to distinguish nodes in the same network in the ER network when $p = 0.4$, while more Priori UMPs were needed to ensure enough shared known nodes among identical nodes in the WS network when $p = 0.05$. The density of the BA network counts for $mv/\binom{v}{2} = 2m/(v-1)$, where v represents the number of nodes in the network. Obviously, the BA network is much sparser, and in this situation more Priori UMPs are required to ensure that the correct UMPs share enough known friends in the sparser networks. In other words, the smaller the m, the sparser the network, and the more Priori UMPs are required.

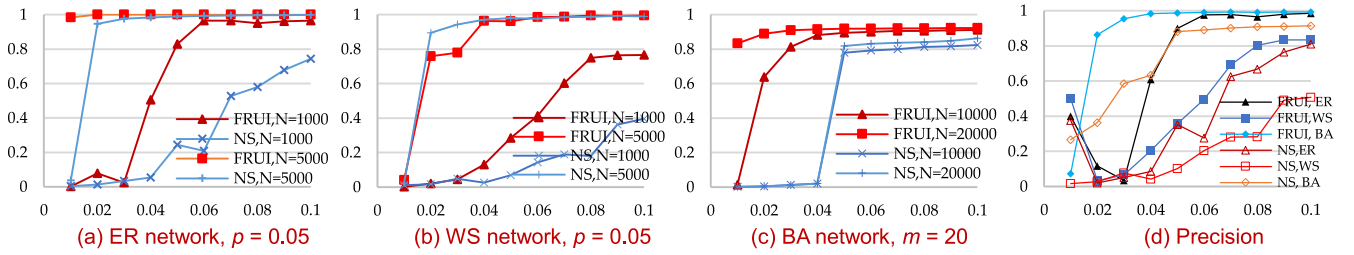


Fig. 6. Comparison between FRUI and NS in synthetic networks. The horizontal ordinates denote the percentage of Priori UMPs. The vertical ordinates are the recall rate in (a), (b) and (c) and precision in (d). In (d), $N = 1000$ in ER and WS networks, and $N = 10{,}000$ in BA networks, where N represents the number of nodes in the networks.

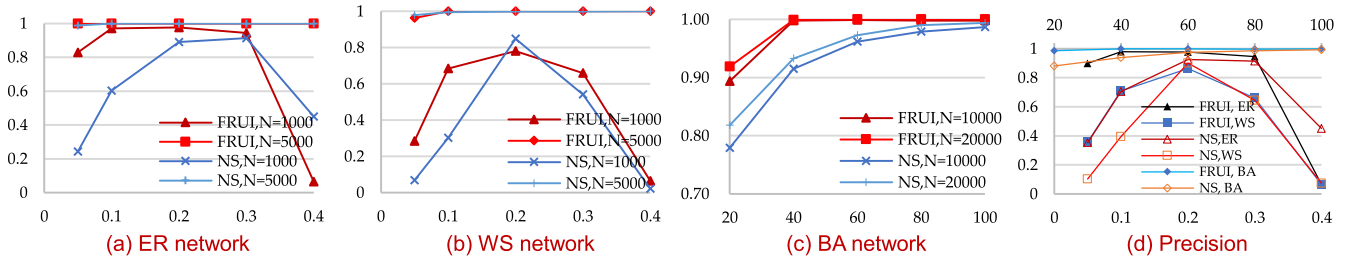


Fig. 7. Comparison between FRUI and NS in synthetic networks. The percentage of Priori UMPs is 5 percent. The horizontal ordinates denote p in (a) and (b), and m in (c). In (d), the horizontal ordinate is p in ER and WS networks, and m in BA network. The vertical ordinates are the recall rate in (a), (b) and (c) and precision in (d). In (d), $N = 1{,}000$ in ER and WS networks, and $N = 10{,}000$ in BA networks.

TABLE 5
Networks of the Ground Truth Dataset

Network	Nodes	Edges	Average Degree
Sina Microblog	1.17 M	1.9 M	3.2
RenRen	5.5 M	14.6 M	5.3

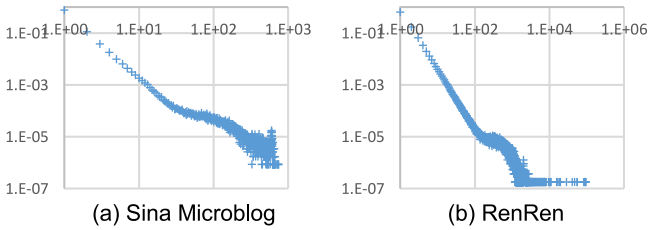


Fig. 8. Degree distributions of networks in ground truth dataset. Both Sina Microblog and RenRen follow a power-law distribution.

6.2 Social Media Network Experiments

In this section, we use ground truth datasets to evaluate the user identification resolution. In order to verify FRUI in different types of SMNs, we collected data from two heterogeneous SMNs: Sina Microblog and RenRen. The Sina Microblog dataset was captured from the Sina Microblog search page, while the RenRen dataset was directly obtained from its Open API. As shown in Table 5, the Sina Microblog dataset consisted of 1.17 million users and 1.9 million friend relationships, and each user had an average of 3.2 friends. The RenRen dataset was comprised of 5.5 million nodes and 14.6 million edges, and each user had an average of 5.3 friends. Therefore, the RenRen dataset was much denser than Sina Microblog's. Fig. 8 illustrates the degree distributions of the two graphs. Clearly, they are scale-free networks [27].

To evaluate the performance of FRUI in real-world datasets, we generated a series of controlled datasets. We selected a pair of subgraphs from both SMNs, and each had over 50,000 nodes. To achieve various degrees of node and edge overlap, we added graphs with different levels of noise. We introduced the Jaccard Coefficient to measure the degree of node/edge overlap,

$$overlap(X, Y) = \frac{|X \cap Y|}{|X \cup Y|}, \tag{12}$$

where *overlap*, X and Y denote the degree of node/edge overlap and the node/edge set of the two graphs, respectively. Thus, when each of the two graphs shares 2/3 of its nodes/edges, the node/edge overlap is 0.5. The degree of edge overlap was limited by the relationships between the overlapped nodes.

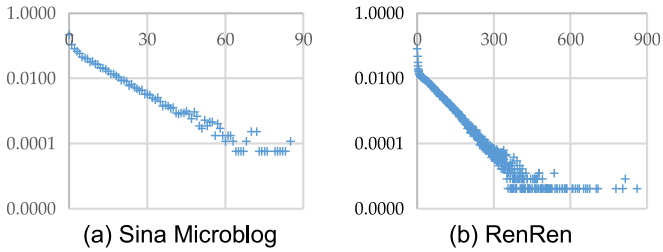


Fig. 10. Distribution of the number of shared neighbors of two connected users in Sina Microblog and Renren.

In the experiments, we randomly chose a number of shared nodes as Priori UMPs. Then we executed user identification in both NS and FRUI. We increased the percentage of Priori UMPs in all UMPs from 0.01 to 0.1 by 0.01. Since the average degrees of both Sina Microblog and RenRen are fairly low, only the nodes with no less than θ neighbors were selected as overlap nodes. To check the performance of FRUI, we increased θ from 20 to 100 by 20.

Fig. 9a compares the recall rates of FRUI and NS with $\theta = 80$. FRUI identified around 50 percent UMPs with 5 percent Priori UMPs, while NS returned no more than 40 percent UMPs with 10 percent Priori UMPs in the Sina Microblog dataset. FRUI also identified many more UMPs in the Renren dataset. It is apparent that FRUI performed much better. Fig. 9b also shows that FRUI performs better in precision than NS in both datasets. Fig. 9c shows that FRUI is less costly than NS in terms of running time, which stands for the elapsed time during the identification process. Results show that FRUI is more efficient in practice, which is consistent with the theoretical analysis. Fig. 9d compares the recall rates of NS and FRUI with 8 percent Priori UMPs in both datasets generated from Sina Microblog and Renren. Results show that FRUI returned many more UMPs than NS in all scenarios in both datasets. In the Sina Microblog dataset, the recall rate of FRUI is around 0.5, which is much larger than that of NS, at about 0.3. The same trend occurred in datasets produced by the RenRen. This indicates that FRUI has more capacity to find UMPs. Fig. 10 indicates that the distributions of the number of the number of shared neighbors between any two connected users in the Sina Microblog and Renren follow an exponential distribution. Most users have much lower degrees in real-world datasets, so that most connected users share a few common users. Those low degree users introduce noise for the NS.

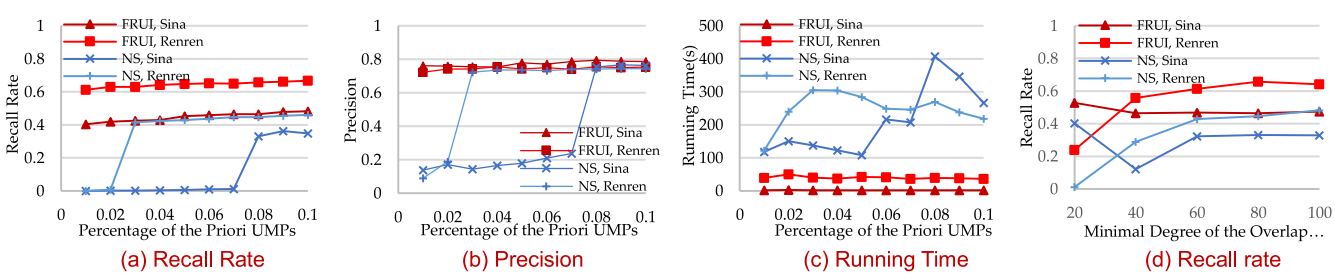


Fig. 9. Comparison between FRUI and NS in Sina and Renren datasets. Node overlap is 33 percent, and edge overlap is 33 percent. In (a), (b) and (c), the minimal degree of the overlap nodes is 80. In (d), there are 8 percent Priori UMPs.

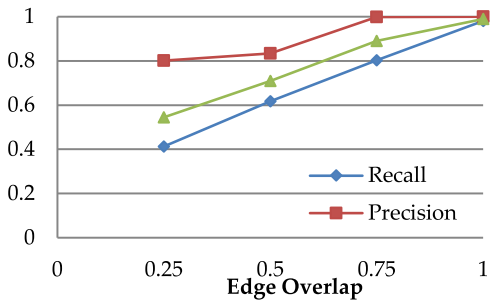


Fig. 11. Effect of noise. The node overlap is 33 percent, and the percentage of Priori UMPs in all the UMPs is 3 percent.

Figs. 9a and 9c further reveals that both FRUI and NS yielded better results in the RenRen subgraphs than in the Sina Microblog subgraphs. This is because both Sina Microblog and RenRen are sparse SMNs. Analogous to discussions the synthetic networks in Section 5.1 and consistent with the analysis in Section 4.2, the denser network structure can benefit both FRUI and NS in those sparser SMNs.

To study the effects of noise, we conducted experiments shown in Fig. 11. It is evident that as the noise increases, the evaluation indices decrease. Nonetheless, FRUI still identified a large volume of identical users in the noisy environment.

We also evaluated how well FRUI identified users across Sina Microblog and RenRen by conducting three groups of experiments. In each experiment, we selected a pair of subgraphs by starting with the identical users and extracting two-layer friends using a breadth-first search. Then we manually labeled 150 users as Priori UMPs and performed FRUI and NS. Since the exact number of identical users is unknown, only the precision was compared. We randomly chose 300 identified UMPs to check the precision. Table 6 illustrates the empirical results. Both FRUI and NS identified a portion of the total nodes, since most users have only one neighbor. However, FRUI returned many more UMPs and obtained much higher precisions in all three experiments. These findings reveal that FRUI is much more proficient for recognizing identical users across Sina Microblog and RenRen.

TABLE 6
Comparison of FRUI and NS in Identifying Users across Sina Microblog and RenRen

# pair of Subgraphs (Nodes)			Identified UMPs		Precision	
			FRUI	NS	FRUI	NS
1	Sina	7,926	1,962	691	0.453	0.203
	RenRen	26,422				
2	Sina	7,131	1,645	598	0.427	0.173
	RenRen	24,052				
3	Sina	7,733	1,734	713	0.430	0.217
	RenRen	24,893				

7 Conclusions

This study addressed the problem of user identification across SMN platforms and offered an innovative solution. As a key aspect of SMN, network structure is of paramount importance and helps resolve de-anonymization user identification tasks. Therefore, we proposed a uniform network structure-based user identification solution. We also developed a novel friend relationship-based algorithm called FRUI. To improve the efficiency of FRUI, we described two propositions and addressed the complexity. Finally, we verified our algorithm in both synthetic networks and ground-truth networks.

Results of our empirical experiments reveal that network structure can accomplish important user identification work. Our FRUI algorithm is simple, yet efficient, and performed much better than NS, the existing state-of-art network structure-based user identification solution. In scenarios when raw text data is sparse, incomplete, or hard to obtain due to privacy settings, FRUI is extremely suitable for cross-platform tasks.

Moreover, our resolution can be easily applied to any SMNs with friendship networks, including Twitter, Facebook and Foursquare. It can also be extended to other studies in social computing with cross-platform problems such as targeted marketing [40], information retrieval [41], collaborative filtering [42], sentiment analysis [43] and more. In addition, since only the Adjacent Users are involved in each iteration process, our method is scalable and can be easily applied to large datasets and online user identification applications.

Identifying anonymous users across multiple SMNs is challenging work. Therefore, only a portion of identical users with different nicknames can be recognized with this method. This study built the foundation for further studies on this issue. Ultimately, it is our hope that a final approach can be developed to identify *all* identical users with different nicknames. Other user identification methods can be applied simultaneously to examine multiple SMN platforms. These methods are complementary and not mutually exclusive, since the final decision may rely on human user's involvement. Therefore, we suggest using these methods synergistically and considering strengths and weaknesses for the best results.

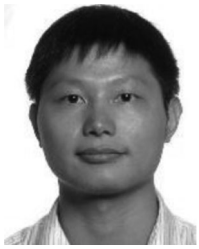
Acknowledgments

This work was supported by the Fundamental Research Funds for the Central Universities, Research Funds of Renmin University of China (10XNI029), the Natural Science Foundation of Beijing under grant no. 4132067, the Natural Science Foundation of China under grant nos. 71531012 and 71271211, and the Beijing Higher Education Young Elite Teacher Project under grant no. YETP1660. The authors highly appreciated the valuable comments of the anonymous reviewers. They are also grateful to Chao Feng for his help on labelling the identical users.

References

[1] Wikipedia. (2014). Twitter [Online]. Available: http://en.wikipedia.org/wiki/Twitter
[2] Xinhuanet. (2014). Sina Microblog Achieves over 500 Million Users [Online]. Available: http://news.xinhuanet.com/tech/2012–02/29/c_122769084.htm
[3] D. Perito, C. Castelluccia, M. A. Kaafar, and P. Manils, "How unique and traceable are usernames?" in *Proc. 11th Int. Conf. Privacy Enhancing Technol.*, 2011, pp. 1–17.
[4] J. Liu, F. Zhang, X. Song, Y. I. Song, C. Y. Lin, and H. W. Hon, "What's in a name?: An unsupervised approach to link users across communities," in *Proc. 6th ACM Int. Conf. Web Search Data Mining*, 2013, pp. 495–504.

[5] R. Zafarani and H. Liu, "Connecting corresponding identities across communities," in *Proc. 3rd Int. ICWSM Conf.*, 2009, pp. 354–357.
[6] R. Zafarani and H. Liu, "Connecting users across social media sites: a behavioral-modeling approach," in *Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2013, pp. 41–49.
[7] A. Acquisti, R. Gross, and F. Stutzman, "Privacy in the age of augmented reality," in *Proc. Nat. Acad. Sci.*, 2011, pp. 36–53, Available: https://www.usenix.org/legacy/events/sec11/tech/slides/acquisti.pdf
[8] T. Iofciu, P. Fankhauser, F. Abel, and K. Bischoff, "Identifying users across social tagging systems," in *Proc. 5th Int. AAAI Conf. Weblogs Social Media*, 2011, pp. 522–525.
[9] M. Motoyama and G. Varghese, "I seek you: searching and matching individuals in social networks," in *Proc. 11th Int. Workshop Web Inf. Data Manage.*, 2009, pp. 67–75.
[10] O. Goga, D. Perito, H. Lei, R. Teixeira, and R. Sommer, "Large-scale correlation of accounts across social networks," University of California at Berkeley, Berkeley, California, Tech. Rep. TR-13-002, 2013.
[11] K. Cortis, S. Scerri, I. Rivera, and S. Handschuh, "An ontology-based technique for online profile resolution," in *Proc. 5th Int. Conf. Social Informat.*, 2013, pp. 284–298.
[12] F. Abel, E. Herder, G. J. Houben, N. Henze, and D. Krause, "Cross-system user modeling and personalization on the social web," *User Model. User-Adapted Interaction*, vol. 23, pp. 169–209, 2013.
[13] O. De Vel, A. Anderson, M. Corney, and G. Mohay, "Mining e-mail content for author identification forensics," *ACM Sigmod Rec.*, vol. 30, no. 4, pp. 55–64, 2001.
[14] E. Raad, R. Chbeir, and A. Dipanda, "User profile matching in social networks," in *Proc. 13th Int. Conf. Netw.-Based Inf. Syst.*, 2010, pp. 297–304.
[15] J. Vosecky, D. Hong, and V. Y. Shen, "User identification across multiple social networks," in *Proc. 1st Int. Conf. Netw. Digital Technol.*, 2009, pp. 360–365.
[16] P. Jain, P. Kumaraguru, and A. Joshi, "@ i seek 'fb. me': Identifying users across multiple online social networks," in *Proc. 22nd Int. Conf. World Wide Web Companion*, 2013, pp. 1259–1268.
[17] P. Jain and P. Kumaraguru, "Finding Nemo: searching and resolving identities of users across online social networks," arXiv preprint arXiv:1212.6147, 2012.
[18] R. Zheng, J. Li, H. Chen, and Z. Huang, "A framework for authorship identification of online messages: Writing style features and classification techniques," *J. Amer. Soc. Inf. Sci. Technol.*, vol. 57, no. 3, pp. 378–393, 2006.
[19] M. Almishari and G. Tsudik, "Exploring linkability of user reviews," in *Proc. 17th Eur. Symp. Res. Comput. Security*, 2012, pp. 307–324.
[20] X. Kong, J. Zhang, and P. S. Yu, "Inferring anchor links across multiple heterogeneous social networks," in *Proc. 22nd ACM Int. Conf. Inf. Knowl. Manage.*, 2013, pp. 179–188.
[21] O. Goga, H. Lei, S. H. K. Parthasarathi, G. Friedland, R. Sommer, and R. Teixeira, "Exploiting innocuous activity for correlating users across sites," in *Proc. 22nd Int. Conf. World Wide Web*, 2013, pp. 447–458.
[22] A. Narayanan and V. Shmatikov, "De-anonymizing social networks," in *Proc. IEEE 30th Symp. Security Privacy*, 2009, pp. 173–187.
[23] S. Bartunov, A. Korshunov, S. Park, W. Ryu, and H. Lee, "Joint link-attribute user identity resolution in online social networks," in *Proc. 6th SNA-KDD Workshop*, 2012.
[24] N. Korula and S. Lattanzi, "An efficient reconciliation algorithm for social networks," arXiv preprint arXiv:1307.1690, 2013.
[25] P. Erdős and A. Rényi, "On random graphs I," *Publ. Math. Debrecen.*, vol. 6, pp. 290–297, 2010.
[26] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," Nature, vol. 393, no. 6684, pp. 440–442, 1998.
[27] A. L. Barabasi and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999.
[28] L. Backstrom, C. Dwork, and J. Kleinberg, "Wherefore art thou r3579x?: Anonymized social networks, hidden patterns, and structural steganography," in *Proc. 16th Int. Conf. World Wide Web*, 2007, pp. 181–190.
[29] B. Zhou and J. Pei, "Preserving privacy in social networks against neighborhood attacks," in *Proc. 24th IEEE Int. Conf. Data Eng.*, 2008, pp. 506–515.
[30] B. Zhou and J. Pei, "The k-anonymity and l-diversity approaches for privacy preservation in social networks against neighborhood attacks," *Knowl. Inf. Syst*, vol. 28, no. 1, pp. 47–77, 2011.
[31] M. Hay, G. Miklau, D. Jensen, and D. Towsley, "Resisting structural identification in anonymized social networks," in *Proc. of the 34th Int. Conf. Very Large Databases*, 2008, pp. 102–114.
[32] K. Liu and E. Terzi, "Towards identity anonymization on graphs," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2008, pp. 93–106.
[33] X. Ying and X. Wu, "Randomizing social networks: A spectrum preserving approach," in *Proc. SIAM Int. Conf. Data Mining*, 2008, pp. 739–750.
[34] Wikipedia. (2014). Social Media [Online]. Available: http://en.wikipedia.org/wiki/Social_media
[35] N. B. Ellison, "Social network sites: Definition, history, and scholarship," *J. Comput. Mediated Commun.*, vol. 13, no. 1, pp. 210–230, 2007.
[36] Q. Xuan and T. Wu, "Node matching between complex networks," *Phys. Rev. E*, vol. 80, no. 2, p. 026103, 2009.
[37] Compete.com. (2014). Connecting the Social Graph: Member Overlap at OpenSocial and Facebook [Online]. Available: http://blog.compete.com/2007/11/12/connecting-the-social-graph-member-overlap-at-opensocial-and-facebook
[38] M. Balduzzi, C. Platzer, T. Holz, T. E. Kirda, D. Balzarotti, and C. Kruegel, "Abusing social networks for automated user profiling," in *Proc. 13th Int. Conf. Recent Adv. Intrusion Detection*, 2010, pp. 422–441.
[39] Y. Xiao, M. Xiong, W. Wang, and H. Wang, "Emergence of symmetry in complex networks," *Phys. Rev. E*, vol. 77, no. 6, p. 066108, 2008.
[40] X. Qian, H. Feng, G. Zhao, and T. Mei, "Personalized recommendation combining user interest and social circle," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 7, pp. 1763–1777, Jul. 2014.
[41] G. Kazai and N. Milic-Frayling, "Trust, authority and popularity in social information retrieval," in *Proc. 17th ACM Conf. Inf. Knowl. Manage.*, 2008, pp. 259–266.
[42] I. Konstas, V. Stathopoulos, and J. M. Jose, "On social networks and collaborative recommendation," in *Proc. 32nd Int. Conf. Res. Develop. Inf. Retrieval*, 2009, pp. 195–202.
[43] S. Tan, Y. Li, H. Sun, Z. Guan, X. Yan, J. Bu, C. Chen, and X. He, "Interpreting the public sentiment variations on twitter," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 5, pp. 1158–1170, May 2014.



Xiaoping Zhou received the BE and ME degrees in computer engineering from Beijing Information Science and Technology University in 2006 and 2009, respectively. He is currently working toward the PhD degree in the Department of Computer Science, Renmin University of China, and is an assistant professor in Beijing University of Civil Engineering and Architecture. His research interests include web mining and social computing.

Xun Liang received the BSc and PhD degrees in computer engineering from Tsinghua University, Beijing, China, in 1989 and 1993, respectively, and the MSc degree in economics and operations research from Stanford University, Stanford, CA, in 1999. He was a postdoctoral fellow at the Institute of Computer Science and Technology, Peking University, from 1993 to 1995, and the Department of Computer Engineering, University of New Brunswick, from 1995 to 1997. He was a software architect or CTO leading over 10 intelligent information products in California from 2000 to 2007 and as an associate professor at the Institute of Computer Science and Technology, Peking University from 2005 to 2009. He is currently a professor in the Department of Computer Science, Renmin University of China. His research interests include machine learning, web mining, and social computing. He is a senior member of the IEEE.

Haiyan Zhang received the BSc degree in computer application from Ningxia University in 1998, and the ME degree in computer software and theory from Shanghai Jiao Tong University in 2005. She is currently working toward the PhD degree in the Department of Computer Science, Renmin University of China. Her research interests include social computing and data mining.

Yuefeng Ma received the BE degree in computer science from Xi'an Jiaotong University, China, in 1997, and the ME degree in management science from Shandong Normal University, China, in 2006. He is currently working toward the PhD degree in the Department of Computer Science, Renmin University of China. His research interests include machine learning, artificial intelligence, and social computing.