Hindawi
Wireless Communications and Mobile Computing
Volume 2020, Article ID 8849536, 10 pages
https://doi.org/10.1155/2020/8849536

Research Article

Trust-Based Missing Link Prediction in Signed Social Networks with Privacy Preservation

Huaizhen Kou,[1] Fan Wang,[1] Chao Lv,[2] Zhaoan Dong,[1] Wanli Huang,[1] Hao Wang,[3] and Yuwen Liu[1]

[1]*School of Computer Science, Qufu Normal University, Rizhao, China*
[2]*China Telecom Smart Home Competence Center, E-Surfing Smart Home Technology Co., Ltd, Nanjing, China*
[3]*Department of Computer Science, Norwegian University of Science and Technology, Trondheim, Norway*

Correspondence should be addressed to Zhaoan Dong; dzan@qfnu.com and Hao Wang; hawa@ntnu.no

Received 28 July 2020; Revised 13 September 2020; Accepted 21 October 2020; Published 16 November 2020

Academic Editor: Zhili Zhou

With the development of mobile Internet, more and more individuals and institutions tend to express their views on certain things (such as software and music) on social platforms. In some online social network services, users are allowed to label users with similar interests as "trust" to get the information they want and use "distrust" to label users with opposite interests to avoid browsing content they do not want to see. The networks containing such trust relationships and distrust relationships are named signed social networks (SSNs), and some real-world complex systems can be also modeled with signed networks. However, the sparse social relationships seriously hinder the expansion of users' social circle in social networks. In order to solve this problem, researchers have done a lot of research on link prediction. Although these studies have been proved to be effective in the unsigned social network, the prediction of trust and distrust in SSN has not achieved good results. In addition, the existing link prediction research does not consider the needs of user privacy protection, so most of them do not add privacy protection measures. To solve these problems, we propose a trust-based missing link prediction method (TMLP). First, we use the simhash method to create a hash index for each user. Then, we calculate the Hamming distance between the two users to determine whether they can establish a new social relationship. Finally, we use the fuzzy computing model to determine the type of their new social relationship (e.g., trust or distrust). In the paper, we gradually explain our method through a case study and prove our method's feasibility.

1. Introduction

With the development of the Internet, more and more individuals or organizations tend to communicate and interact on the network platform. Through social platforms, people can not only share their own feelings about different products but also express their views on others, which greatly enriches people's social activities. However, the rapid development of social platforms has filled them with too many useless or false information and accounts. In order to quickly and easily browse the content for interested, users usually add users who have common interests to the "trust list." At the same time, in order to avoid browsing the content not interested, users usually add users with opposite interests to the "distrust list." For example, $user_1$ and $user_2$ have similar interests, while $user_1$ and $user_3$ have some conflict in a certain area. So, on Twitter, $user_1$ may follow $user_2$ and add $user_3$ to the blacklist. By capturing the trust and distrust relationships between users, we can build a signed social network.

Through the trust/distrust relationships in the signed social network, we can not only know which users the target user has social relationships with but also know what kind of attitude the target user adopts towards these users. However, the social relationships in a social network are too sparse, which seriously hinders the expansion of the user's social circle and the further development of the social platform. Therefore, it has become necessary to help users discover more new friends or trusted users. Fortunately, users have left a large amount of historical behavior data (e.g., user's rating and comments) on social platforms, which provides

favorable conditions for evaluating the trust relationships and similarity of preferences between different users.

However, the approach to finding potential friends through shared interests still faces many challenges. First of all, existing methods focus more on how to predict the trust relationships between users and ignore the role of distrust relationships in the social network. Secondly, the existing methods do not consider how to protect the user's private information. Rating information is a piece of very important private information for users. Once the rating information is disclosed, users may be affected by targeted marketing. For example, some criminals learned about your rating information and fabricated or disseminated data, infringing on the user's private information. In response to these challenges, we propose a trust-based missing link prediction method to find new trust/distrust social relationships for users in social networks.

In general, we have two contributions in this paper:

(1) In this paper, we use simhash technology to find users who may establish a social relationship with the target user. This technology is not sensitive to the historical data of user behavior, which effectively protects the user's private information and greatly reduces the calculation range

(2) We use the fuzzy computing model to predict the types of social relationships that may be established between users (that is, to form a signed social network)

The rest of this paper is organized as follows. Related work is introduced in Section 2. In Section 3, we introduced research motivation. Section 4 introduced the simhash-based link prediction method we proposed in detail. In Section 5, we conducted a case study to prove that our method is feasible. Finally, we summarize this paper in Section 6.

2. Related Work

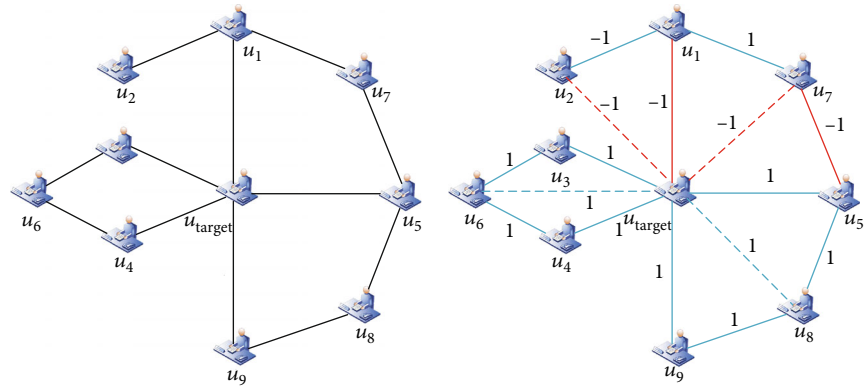
2.1. Link Prediction. As an effective method to solve network sparsity, many studies [1] have used link prediction methods to predict missing edges in networks. For example, Qi et al. [2] proposed a web API recommendation method to generate links between compatible web APIs. Naturally, link prediction methods are also used to solve problems in social networks. Zhang et al. [3] used the network structure and user information to efficiently predict future friendships between users, thereby improving customer loyalty and user experience. Kutty et al. [4] are committed to predicting new social relationships between two different sets of users. For example, there are two user collections: teachers and students. Kutty et al. will predict a new social relationship between a teacher and a student. Wang et al. [5] integrated the cyber, physical, and social spaces together and proposed a distributed method with its incremental calculation for big data in cyberphysical social systems and then used big data of network physical society to calculate tensor and optimize the model. Yang et al. [6] proposed an online social network recommendation system based on Bayesian inference, which attempts to help users establish social relationships with users with similar ratings. In addition, Zhou et al. [7] propose a coarse-to-fine feature matching scheme using both global and local CNN features for real-time near-duplicate image detection. Zhou et al.'s method has some creative inspirations for finding the social relationship between users through feature matching.

2.2. Signed Social Network. Although many people have studied how to solve the sparse problem in social networks, they only focus on unsigned social networks. Fortunately, a growing number of researchers have realized that social relationships between users are signed and have studied trust/distrust in social networks. Xu et al. [8] applied trust relationships to edge computing of social networks. Beigi et al. [9] distinguished unsigned social networks from signed social networks and used three social science theories to study the problem of predicting social relationships in SSN. Wen et al. [10] studied the differences in people's behaviors when they tended to believe and not believe and confirmed their impact on the spread of information on social media. Xu et al. [11] were devoted to using trust relationships for vehicle internet video monitoring offloading service. In addition, Li et al. [12] studied the community diversified influence maximization (CDIM) problem and solved a series of computing challenges in social networks.

2.3. Privacy Protection. In addition, privacy protection is also a research hotspot in related fields, attracting many scholars to participate in the research. Liu et al. [13] proposed an outsource real-time route planning (or2p) scheme, which can protect user trajectory data in route planning. Zhong et al. [14] proposed a multidimensional quality ensemble-driven recommendation approach named RecLSHTOPSIS based on LSH and TOPSIS (Technique for Order Preference by Similarity to Ideal Solution) techniques to protect multidimensional user Qos privacy data in mobile edge computing. Chi et al. [15] proposed LSH-based recommender systems with privacy protection. Xu et al. [16] proposed a blockchain-powered crowdsourcing method considering privacy preservation in a mobile environment. Although the above research is very effective, there is not much research on protecting user private information in social networks. Qi et al. [17] proposed a kind based on the classic Locality-Sensitive Hashing (LSH) technique to protect privacy data in a smart city. In addition, Zhou et al. [18] propose a novel coverless steganographic approach without any modification for transmitting a secret color image. This method has some enlightenment for protecting users' private information.

3. Research Motivation

As shown in Figure 1, each node in the figure represents users in the social network, and each edge represents the social relationships between users. In Figure 1, u_{target} represents the target user and the other nodes represent users associated with u_{target}. In the social network shown in Figure 1(a), the black lines represent the social relationships that exist between users, but it is not known whether the relationship between users is trust or distrust. When the social relationship contains the information of whether the user trusts or not, it constitutes



(a) A social network (b) A signed social network

Figure 1: An intuitive example of our motivation.

the signed social network as shown in Figure 1(b). The blue lines represent the social relationships of trust, the red lines represent the social relationships of distrust, the solid lines represent the existing social relationships, and the dashed lines represent the possible social relationships.

As can be seen from Figure 1, the existing social relationships between users are relatively sparse, but there are still many potential social relationships waiting to be explored in the social network. In order to increase the number of edges in a social network, we need to calculate the possibility of establishing a social relationship between users and calculate the trust/distrust value between users. However, we face many challenges in this process. First of all, there are tens of thousands of users in social networks, and it takes a lot of computing power to calculate the trust/distrust relationships between any two users. This can place a huge burden on the server, prolong computing time, and ultimately create a bad user experience. Secondly, users generally care about whether their private information will be disclosed by social platforms. In fact, the user's rating information can accurately reflect the user's interests and hobbies. However, obtaining the user's rating information without permission is usually considered to be offensive, and at the same time, users will be harassed by marketing advertisements. However, in the process of calculating whether users can establish new social relations, users' privacy will be frequently accessed, which will easily lead to user privacy disclosure. Therefore, we need to design a method that can effectively protect the user's private information and significantly reduce the amount of calculation while predicting the social relationship between users.

4. Trust-Based Missing Link Prediction Method

There are both trust and distrust relationships in social networks. Most of the existing link prediction methods in the field of social networks only predict the trust relationship between users and ignore the distrust relationship. In fact, distrust relationship is also crucial in social networks, so we propose a new link prediction method named the trust-based missing link prediction method (TMLP), which can predict both trust and distrust relationships. In addition, considering the user's demand for privacy protection, the TMLP method can also effectively protect the user's private information from being disclosed. The principle of the TMLP method will be explained in detail below.

Step 1. Build a hash index for each user.

Hash is a common verification [19] and mapping technology, and simhash is a better mapping technology in hash. It is well known that the principle of the simhash method [20] is that the more similar the items that two users interact with, the more similar their simhash values are. From this, we can see that if we want to find users who may have a social relationship with the target users, we only need to compare their simhash values. This subsection will explain how to create a simhash index for each user based on their behavior history.

In Figure 2, u_{target} represents the target user who needs to establish social relationships. $I = \{i_1, \cdots, i_n\}$ represents a collection of all items, and users in a social network interact with items in I. $\widehat{I} = \{i_1, \cdots, i_m\}$ is a collection of items that u_{target} has interacted with. Simhash technology can map the interaction history of u_{target} into a one-dimensional vector, which is represented by $H(u)$.

First, we set up for each item in I a random r-dimensional vector consisting only of "0" and "1," in which $r = \lceil \log_2 n \rceil$ ($\lceil x \rceil$ is taking the round number in the x direction, e.g., $\lceil 4.4 \rceil = 5$). In the example shown in Figure 2, $r = 6$. According to Formula (1), we form the items the user has interacted with into an $r * m$ matrix $h_1(u_{target}) = (V_1, \cdots, V_m)$:

$$V_j = \begin{cases} v_j, & \text{if } u_{target} \text{ invoked } i_j \text{ before,} \\ \text{Null}, & \text{if } u_{target} \text{ never invoked } i_j \text{ before.} \end{cases} \quad (1)$$

Next, as shown in Step (1) in Figure 2, we delete the null value in $h_1(u_{target})$ and replace the "0" in it with "-1" to get a new matrix $h_2(u_{target})$. Next, we take the sum of the columns of $h_2(u_{target})$, and we get an r-dimensional vector $h_3(u_{target})$ as shown in Step (2) in Figure 2. Finally, as shown in Step

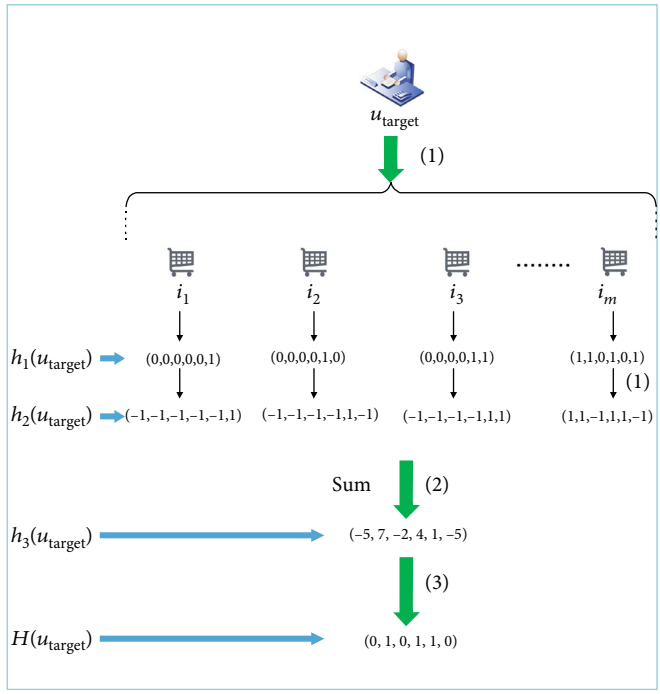


Figure 2: An example of simhash.

(3) in Figure 2, we set values greater than "0" in $h_3(u_{target})$ to "1" and values less than or equal to "0" to "0," to obtain the user's simhash index $H(u_{target})$. According to the simhash theory [21], $H(u_{target})$ can be regarded as the index of user u_{target}. At this point, we can create a simhash index for each user through this method.

Step 2. Build the set of users who may establish social relationships with the target user.

In the previous step, we created a simhash index $H(u_i)$ for each user based on their behavior history. Next, we need to determine which users might have social relationships with u_{target}.

We first calculate the Hamming distance between $H(u_{target})$ and $H(u_i)$ $(1 \le i \le n)$, which is represented as $D(H(u_{target}), H(u_i))$. Specifically speaking, assume $H(u_{target})$ and $H(u_i)$ are represented by the r-dimensional vectors $(v_{target-1}, \cdots, v_{target-r})$ and $(v_{i-1}, \cdots, v_{i-r})$, respectively. Then, $D(H(u_{target}), H(u_i))$ can be calculated by Formula (2), where a_f is the Boolean value calculated by Formula (3). Here, the sign " $\oplus$ " refers to XOR operation:

$$D\left(H\left(u_{target}\right), H(u_i)\right) = \sum a_f (1 \le f \le r), \tag{2}$$

$$a_f = v_{target-f} \oplus v_{i-f} = \begin{cases} 1, & \text{if } v_{target-f} \ne v_{i-f}, \\ 0, & \text{if } v_{target-f} = v_{i-f}, \end{cases} \tag{3}$$

$$\text{if } D\left(H\left(u_{target}\right), H(u_1)\right) = \begin{cases} < \left\lceil \frac{r}{k} \right\rceil, & \text{future link can be predicted,} \\ \ge \left\lceil \frac{r}{k} \right\rceil, & \text{prediction not possible.} \end{cases} \tag{4}$$

The more similar the items that two users interact with, the smaller the Hamming distance, and the more likely they are to establish a social relationship. In Formula (4), if the Hamming distance between the target user u_{target} and user u_i is smaller than $\lceil r/k \rceil$ $(2 \le k \le r)$, u_i can be regarded as a possible linked user to u_{target}, and u_i is put into the possible linked users (PLU (u_{target})) set of u_{target}. In addition, the pseudocode used to build a possible set of users linked to the target user is specified in Algorithm 1. With the simhash method, we build an index for each user that is insensitive to their historical data. Therefore, the simhash method can also effectively protect user privacy. In short, on the premise of effectively protecting users' privacy, we have found users who can establish social relations for target users.

Step 3. Calculate whether social relationships are positive or negative.

Now that we know which users might have social relationships with their target users, that is not enough. If user A and user B have seen the same four movies, but user A likes movies 1 and 2, while user B likes movies 3 and 4, we cannot think that user A and user B trust each other. So next, we need to predict the types of these social relations (new links). Based on the methods of [22, 23], we propose a trust-distrust fuzzy computing method based on the user preference similarity in this section.

Require: the simhash index of the target user: $H(u_{target})$ & the simhash index of each user $u_i(u_i \in U)$: $H(u_i)$
Ensure: each target user sets PLU (u_{target})
Let PLU $(u_{target}) = \Phi$
while each $u_i \in U$ **do**
  if $D(H(u_{target}), H(u_i)) < \lceil r/k \rceil$ **then**
    $D(H(u_{target}), H(u_i)) = \sum a_f (1 \le f \le r)$
    if $v_{target-f} = v_{i-f}$ **then**
      $a_f = v_{target-f} \oplus v_{i-f} = 0$
    else
      $a_f = v_{target-f} \oplus v_{i-f} = 1$
    end if
    enqueue u_i into PLU(u_{target})
    update PLU(u_{target})
  end if
end while

Algorithm 1. PLU (u_{target}).

The calculation of preference similarity based on the user's rating of items is actually the construction of a weighted social network [24], and the sign and sizes of weights on social relations are used to judge whether users have a trusting relationship or a distrustful relationship. Users usually score items in the range of 1-5. For 3-5 points, we can think that the user likes the item, while for 1-2 points, we can think that the user does not like the item. If two users have similar ratings on an item, we can assume that their preferences are similar. Therefore, the user's rating of an item can be regarded as a fuzzy variable.

Therefore, we adopt the half triangular membership function [25] defined in Formula (5). This half triangular membership function represents the continuity of fuzzy set Z from the minimum value (min) to the maximum value (max), in which $R_{(u_i,I_k)}$ refers to the rating of the k_{th} item made by the i_{th} user:

$$\mu_Z(I_k) = \begin{cases} 0, & R_{u_i}, I_k = \min, \\ \dfrac{R_{u_i}, I_k - \min}{\max - \min}, & \min < R_{u_i},\ I_k < \max, \\ 1, & R_{u_i}, I_k = \max. \end{cases} \tag{5}$$

Based on the fuzzy set Z, the items are classified into two types: helpful item (H) and the unhelpful item (NH):

$$H = \{I_k : \mu_Z(I_k) > 0.5\}, \tag{6}$$

$$\text{NH} = \{I_k : \mu_Z(I_k) \le 0.5\}. \tag{7}$$

The preference similarity between u_i and u_j is represented as $\text{Pre\_Sim}(u_i, u_j)$ and $\text{Pre\_Non\_Sim}(u_i, u_j)$, while $\text{Pre\_Sim}(u_i, u_j)$ and $\text{Pre\_Non\_Sim}(u_i, u_j)$ are expressed by Formulas (8) and (9), respectively:

$$\text{Pre\_Sim}(u_i, u_j) = \frac{1}{2}\left[\frac{|H_{u_i} \cap H_{u_j}|}{|H_{u_i}|} + \frac{|\text{NH}_{u_i} \cap \text{NH}_{u_j}|}{|\text{NH}_{u_i}|}\right], \tag{8}$$

$$\text{Pre\_Non\_Sim}(u_i, u_j) = \frac{1}{2}\left[\frac{|H_{u_i} \cap \text{NH}_{u_j}|}{|H_{u_i}|} + \frac{|\text{NH}_{u_i} \cap H_{u_j}|}{|\text{NH}_{u_i}|}\right]. \tag{9}$$

The trust and distrust relations between users are represented by $\text{Overall\_Trust}(u_i, u_j)$ and $\text{Overall\_Distrust}(u_i, u_j)$, respectively. Whether the social relationship between two users is a trust relationship or a distrust relationship depends on the relative size of $\text{Overall\_Trust}(u_i, u_j)$ and $\text{Overall\_Distrust}(u_i, u_j)$. If $\text{Overall\_Trust}(u_i, u_j)$ is greater than $\text{Overall\_Distrust}(u_i, u_j)$, the two users are in a trust relationship, and vice versa. The calculation is as follows:

$$\text{Overall\_Trust}(u_i, u_j) = \text{Pre\_Sim}(u_i, u_j), \tag{10}$$

$$\text{Overall\_Distrust}(u_i, u_j) = \text{Pre\_Non\_Sim}(u_i, u_j). \tag{11}$$

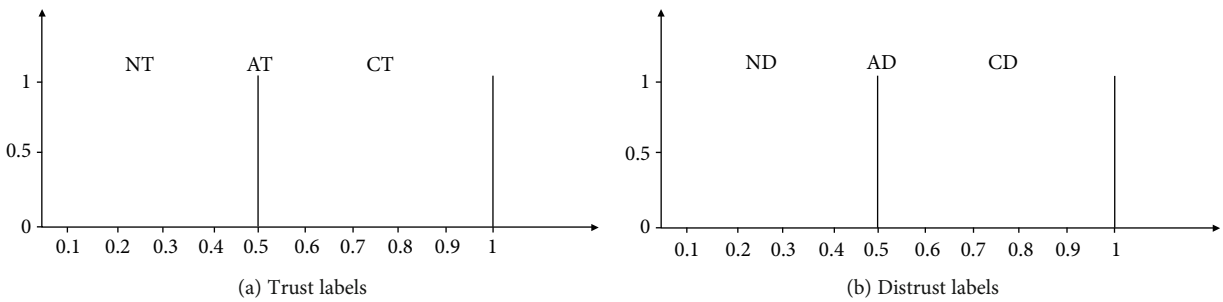
The trust values in Figure 3 are fuzzy into three normal fuzzy sets. The fuzzy sets of trust include CT (complete trust), AT (almost trust), and NT (not trust). Similarly, the untrusted fuzzy sets include CD (complete distrust), ad (almost distrust), and Nd (not distrust). Please note that in a social network, the social relationship between two users is clear. If $\text{Overall\_Trust}(u_i, u_j)$ between the two users is greater than $\text{Overall\_Distrust}(u_i, u_j)$, the two users are considered to trust each other; otherwise, they are regarded as distrustful. The pseudocode for the TMLP method is specified in Algorithm 2.

5. A Case Study

In this section, we will demonstrate the process of the TMLP method through a case study. As shown in Figure 4, there are nine users in the social network forming a user set $U = \{u_{\text{target}}, u_1, \cdots, u_8\}$. Among them, u_{target} is the target user. In Figure 4, the connection between users represents the social relationship, where the solid line represents the existing social relationship, the dashed line represents the potential social relationship, the blue line represents the trust relationship, and the red line represents the distrust relationship. Then, we selected 16 movies to form the movie set $M = \{m_1, \cdots, m_{16}\}$. The record of the user watching the movie is shown in Table 1. 1-5 indicates the rating given by the user, and 0 indicates that the user has not watched the movie.

Step 1. Build a hash index for each user in U.

First, we calculate the hash value of the movie in M to get the set $V = \{v_1, \cdots, v_{16}\}$, and the results are shown in Table 2. Then, according to Formula (1) and user rating records, we can get $h_1(u_i)$ shown in Table 3.



(a) Trust labels (b) Distrust labels

Figure 3: Membership functions for trust and distrust.

Require: u_{target} and PLU (u_{target})
Ensure: $LP(H(u_{target}), H(u_i))$ and New signal social network
  while $u_i \in PLU(u_{target})$ **do**
    $Overall\_Trust(u_{target}, u_i)$ or $Overall\_Distrust(u_{targer}, u_i)$
    if $Overall_{Trust(u_{target}, u_i)} < Overall\_Trust(u_{target}, u_i)$ **then**
      $LP(H(u_{target}), H(u_i)) = -1$
    else
      $LP(H(u_{target}), H(u_i)) = +1$
    end if
  end while

Algorithm 2. TMLP.

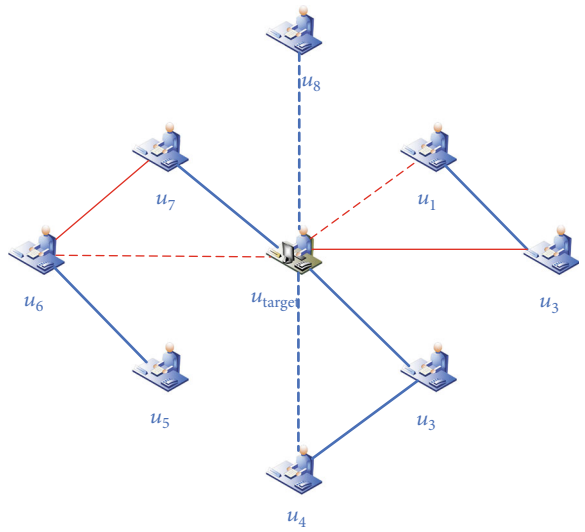


Figure 4: An example of a signed social network.

Next, in vector $h_1(u_i)$, we will delete dimensions with a null value and replace "0" with "−1." Then, we will obtain a new vector $h_2(u_i)$. Next, in the generated $n * r$ matrix, we calculate the sum of each column and obtain a new vector $h_3(u_i)$. Finally, the positive and negative values were replaced with "1" and "0," respectively, and we will obtain a new vector $H(u_i)$. The new vector $H(u_i)$ is the simhash value of u_i. Formulas (12) and (13) show the process of generating $h_3(u_{\text{target}})$ from $h_2(u_{\text{target}})$, and then $h_3(u_{\text{target}})$ to $H(u_{\text{target}})$.

Table 4 shows the simhash values of all users in U:

$$h_2(u_{\text{target}}) = \begin{bmatrix} -1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ -1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 & -1 & 1 & 1 & 1 \\ -1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 \\ 1 & -1 & -1 & -1 & 1 & 1 & 1 & -1 \\ 1 & -1 & 1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & -1 \\ 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 \end{bmatrix} \Rightarrow h_3(u_{\text{target}}) = [2\ 2\ -2\ 2\ -2\ -4\ 0\ -4], \tag{12}$$

$$h_3(u_{\text{target}}) = [2 \quad 2 \quad -2 \quad 2 \quad -2 \quad -4 \quad 0 \quad -4] \Rightarrow H(u_{\text{target}}) = [1 \quad 1 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0]. \tag{13}$$

Step 2. Build the set of users who may establish a social relationship with the target user.

Now, we need to find the set of users (PLU (u_{target})) who may establish a social relationship with the target user according to the simhash value. We first calculate the

Table 1: Viewing record.

	u_{target}	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8
m_1	5	0	2	0	0	0	0	0	3
m_2	4	1	1	0	5	0	1	3	5
m_3	5	2	3	4	0	0	0	2	4
m_4	5	1	0	5	4	1	2	5	5
m_5	3	3	3	4	3	3	3	4	5
m_6	1	5	4	3	1	5	4	1	3
m_7	2	4	5	1	2	2	5	3	2
m_8	1	4	4	1	2	3	3	2	1
m_9	1	5	5	2	1	3	4	2	1
m_{10}	2	5	1	3	3	5	4	1	0
m_{11}	0	5	0	2	2	2	0	0	0
m_{12}	0	0	0	0	0	1	5	0	0
m_{13}	0	0	0	0	0	0	0	0	4
m_{14}	0	0	0	0	0	0	0	5	0
m_{15}	0	0	0	0	0	0	0	1	0
m_{16}	0	0	1	0	0	0	0	0	0

Table 2: Movie hash.

Movies	Hash
v_1	0 1 1 1 0 0 0 0
v_2	0 1 0 0 0 0 1 1
v_3	1 1 0 1 0 1 1 1
v_4	0 1 0 0 0 0 0 0
v_5	0 0 1 1 0 0 0 0
v_6	1 0 0 1 1 1 1 0
v_7	1 0 0 0 1 1 1 0
v_8	1 0 1 1 1 0 0 1
v_9	1 1 0 0 0 0 1 0
v_{10}	1 1 1 1 1 0 0 0
v_{11}	1 1 1 0 1 1 0 1
v_{12}	1 0 1 0 0 0 0 1
v_{13}	0 0 1 1 0 0 1 1
v_{14}	1 1 1 1 0 0 0 0
v_{15}	1 1 1 0 0 0 1 1
v_{16}	0 1 0 1 1 0 1 0

Hamming distance between $H(u_{target})$ and $H(u_i)$ $(1 \le i \le m)$, which is represented as $D(H(u_{target}), H(u_i))$. The Hamming distance between u_{target} and other users calculated according to Formula (2) is shown in Table 5.

At this time, we set the threshold of Hamming distance to 3; that is, users whose Hamming distance is less than 3 belong to the set PLU (u_{target}).

Table 3: $h_1(u_i)$ for each user.

Users	$h_1(u_i)$
u_{target}	$V_1, V_2, V_3, V_4, V_5, V_6, V_7, V_8, V_9, V_{10}$
u_1	$V_2, V_3, V_4, V_5, V_6, V_7, V_8, V_9, V_{10}, V_{11}$
u_2	$V_1, V_2, V_3, V_5, V_6, V_7, V_8, V_9, V_{10}, V_{16}$
u_3	$V_3, V_4, V_5, V_6, V_7, V_8, V_9, V_{10}, V_{11}$
u_4	$V_2, V_4, V_5, V_6, V_7, V_8, V_9, V_{10}, V_{11}$
u_5	$V_4, V_5, V_6, V_7, V_8, V_9, V_{10}, V_{11}, V_{12}$
u_6	$V_2, V_4, V_5, V_6, V_7, V_8, V_9, V_{10}, V_{12}$
u_7	$V_2, V_3, V_4, V_5, V_6, V_7, V_8, V_9, V_{10}, V_{14}, V_{15}$
u_8	$V_1, V_2, V_3, V_4, V_5, V_6, V_7, V_8, V_9, V_{13}$

Table 4: User simhash.

Users	Simhash
u_{target}	1 1 0 1 0 0 0 0
u_1	1 1 0 0 0 0 0 0
u_2	1 1 0 1 0 0 1 0
u_3	1 1 0 1 1 0 0 0
u_4	1 1 0 0 1 0 0 0
u_5	1 0 1 0 1 0 0 0
u_6	1 0 0 0 0 0 0 0
u_7	1 1 0 1 0 0 1 0
u_8	0 0 0 1 0 0 1 0

Table 5: Hamming distance.

	u_{target}	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8
u_{target}	0	1	1	1	2	4	2	1	3

Step 3. Calculate the similarity between the target user and other users to determine the type of social relationships.

Now that we know which users are able to establish social relationships with target users, we will next determine the types of these new social relationships. First, we use the half triangular membership function defined by Formula (5) to determine the continuity of the fuzzy set Z, as shown in Table 6.

Based on the fuzzy set Z and Formulas (6) and (7), we can classify the movies that users have watched as like (H) and dislike (NH), as shown in Table 7.

According to the movies that users like and the movies they do not like, we can calculate the similarity of preferences between u_{target} and users in PLU (u_{target}), as shown in Tables 8 and 9.

From the preference similarity, we can get the trust value and the distrust value between the two users and finally determine whether the two users trust or distrust, as shown in Tables 10 and 11.

Table 6: User attitude towards movies.

	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}	m_{11}	m_{12}	m_{13}	m_{14}	m_{15}	m_{16}
u_{target}	1	0.75	1	1	0.5	0	0.25	0	0	0.25	Null	Null	Null	Null	Null	Null
u_1	Null	0	0.25	0	0.5	1	0.75	0.75	1	1	1	Null	Null	Null	Null	Null
u_2	0.25	0	0.5	Null	0.5	0.75	1	0.75	1	0	Null	Null	Null	Null	Null	0
u_3	Null	Null	0.75	1	0.75	0.5	0	0	0.25	0.5	0.25	Null	Null	Null	Null	Null
u_4	Null	1	Null	0.75	0.5	0	0.25	0.25	0	0.5	0.25	Null	Null	Null	Null	Null
u_5	Null	Null	Null	0	0.5	1	0.25	0.5	0.5	1	0.25	0	Null	Null	Null	Null
u_6	Null	0	Null	0.25	0.5	0.75	1	0.5	0.75	0.75	Null	1	Null	Null	Null	Null
u_7	Null	0.5	0.25	1	0.75	0	0.5	0.25	0.25	0	Null	Null	Null	1	0	Null
u_8	0.5	1	0.75	1	1	0.5	0.25	0	0	Null	Null	Null	0.75	Null	Null	Null

Table 7: User preference.

Users	Like	Dislike
u_{target}	m_1, m_2, m_3, m_4, m_5	$m_6, m_7, m_8, m_9, m_{10}$
u_1	$m_5, m_6, m_7, m_8, m_9, m_{10}, m_{11}$	m_2, m_3, m_4
u_2	$m_3, m_5, m_6, m_7, m_8, m_9$	m_1, m_2, m_{10}, m_{16}
u_3	$m_3, m_4, m_5, m_6, m_{10}$	m_7, m_8, m_9, m_{11}
u_4	m_2, m_4, m_5, m_{10}	$m_6, m_7, m_8, m_9, m_{11}$
u_5	$m_5, m_6, m_8, m_9, m_{10}$	m_4, m_7, m_{11}, m_{12}
u_6	$m_5, m_6, m_7, m_8, m_9, m_{10}$	m_2, m_4
u_7	$m_2, m_4, m_5, m_7, m_{14}$	$m_3, m_6, m_8, m_9, m_{10}, m_{15}$
u_8	$m_1, m_2, m_3, m_4, m_5, m_6, m_{13}$	m_7, m_8, m_9

Table 8: Pre_Sim(u_{target}, u_i).

	u_{target}	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8
u_{target}	1	0.1	0.3	0.7	0.7	0	0.1	0.7	0.8

Table 9: Pre_Non_Sim(u_{target}, u_i).

	u_{target}	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8
u_{target}	0	0.8	0.6	0.2	0.1	0	0.7	0.2	0.1

Table 10: Trust/distrust value.

	u_{target}	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8
u_{target}	1	-0.7	-0.3	0.5	0.6	0	-0.6	0.5	0.7

Table 11: Trust/distrust relationship.

	u_{target}	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8
u_{target}	T	DT	DT	T	T	Null	DT	T	T

6. Conclusions and Future Work

In this paper, we mainly propose a novel link prediction method (i.e., trust-based missing link prediction (TMLP)) to find missing social relationships in signed social networks and predict possible social relationships. In addition, we also conducted research on how to effectively protect user privacy during the link prediction process. Finally, through a case study, we verified the feasibility of this novel link prediction method. However, there are some shortcomings in our approach. For example, our method does not use only case studies without actual experimental validation. Furthermore, our method does not consider the network delay and energy consumption of social platforms. In the future work, we will carry out a series of experiments to verify our method and make it more convincing. Then, we will consider using an edge computing algorithm [26, 27] to solve the problem of social platform in the Internet, so as to provide better services for users. And we also note that there have been some studies [28] on local feature matching, and we will use the inspirations from these studies for future work.

Data Availability

The research was demonstrated through a case study and therefore did not use publicly available data sets.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This paper is partially supported by the Natural Science Foundation of China (No. 61872219), the Natural Science Foundation of Shandong Province (ZR2019MF001), and the Open Project of State Key Laboratory of Novel Software Technology (No. KFKT2020B08).

References

[1] H. Liu, H. Kou, X. Chi, and L. Qi, "Combining time, keywords and authors information to construct papers correlation graph (s)," *The 31st International Conference on Software Engineering and Knowledge Engineering*, pp. 11–19, 2019.

[2] L. Qi, Q. He, F. Chen, X. Zhang, W. Dou, and Q. Ni, "Data-driven web apis recommendation for building web applications," in *IEEE Transactions on Big Data*, 2020.

[3] Z. Zhang, J. Wen, L. Sun, Q. Deng, S. Su, and P. Yao, "Efficient incremental dynamic link prediction algorithms in social network," *Knowledge-Based Systems*, vol. 132, pp. 226–235, 2017.

[4] S. Kutty, R. Nayak, and L. Chen, "A people-to-people matching system using graph mining techniques," *World Wide Web*, vol. 17, no. 3, pp. 311–349, 2014.

[5] X. Wang, W. Wang, L. T. Yang, S. Liao, D. Yin, and M. J. Deen, "A distributed hosvd method with its incremental computation for big data in cyber-physical-social systems," *IEEE Transactions on Computational Social Systems*, vol. 5, no. 2, pp. 481–492, 2018.

[6] X. Yang, Y. Guo, and Y. Liu, "Bayesian-inference-based recommendation in online social networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 4, pp. 642–651, 2012.

[7] Z. Zhou, K. Lin, Y. Cao, C.-N. Yang, and Y. Liu, "Near-duplicate image detection system using coarse-to-fine matching scheme based on global and local cnn features," *Mathematics*, vol. 8, no. 4, p. 644, 2020.

[8] X. Xu, X. Liu, Z. Xu, F. Dai, X. Zhang, and L. Qi, "Trust-oriented iot service placement for smart cities in edge computing," *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 4084–4091, 2020.

[9] G. Beigi, J. Tang, and H. Liu, "Signed link analysis in social media networks," 2016, http://arxiv.org/abs/1603.06878.

[10] S. Wen, M. S. Haghighi, C. Chen, Y. Xiang, W. Zhou, and W. Jia, "A sword with two edges: propagation studies on both positive and negative information in online social networks,"

IEEE Transactions on Computers, vol. 64, no. 3, pp. 640–653, 2014.

[11] X. Xu, Q. Wu, L. Qi, W. Dou, S.-B. Tsai, and M. Z. A. Bhuiyan, "Trust-aware service offloading for video surveillance in edge computing enabled internet of vehicles," *IEEE Transactions on Intelligent Transportation Systems*, 2020.

[12] J. Li, T. Cai, K. Deng, X. Wang, T. Sellis, and F. Xia, "Communitydiversified influence maximization in social networks," *Information Systems*, no. article 101522, 2020.

[13] Q. Liu, P. Hou, G. Wang, T. Peng, and S. Zhang, "Intelligent route planning on large road networks with efficiency and privacy," *Journal of Parallel and Distributed Computing*, vol. 133, pp. 93–106, 2019.

[14] W. Zhong, X. Yin, X. Zhang et al., "Multi-dimensional quality-driven service recommendation with privacy-preservation in mobile edge environment," *Computer Communications*, vol. 157, pp. 116–123, 2020.

[15] X. Chi, C. Yan, H. Wang, W. Rafique, and L. Qi, "Amplified lsh-based recommender systems with privacy protection," *Concurrency and Computation: Practice and Experience*, 2020.

[16] X. Xu, Q. Liu, X. Zhang, J. Zhang, L. Qi, and W. Dou, "A blockchain-powered crowdsourcing method with privacy preservation in mobile environment," *IEEE Transactions on Computational Social Systems*, vol. 6, no. 6, pp. 1407–1419, 2019.

[17] L. Qi, C. Hu, X. Zhang et al., "Privacy-aware data fusion and prediction with spatialtemporal context for smart city industrial environment," *IEEE Transactions on Industrial Informatics*, 2020.

[18] Z. Zhou, Y. Mu, and Q. J. Wu, "Coverless image steganography using partial-duplicate image retrieval," *Soft Computing*, vol. 23, no. 13, pp. 4927–4938, 2019.

[19] Y. Xu, J. Ren, Y. Zhang, C. Zhang, B. Shen, and Y. Zhang, "Blockchain empowered arbitrable data auditing scheme for network storage as a service," *IEEE Transactions on Services Computing*, vol. 13, no. 2, pp. 289–300, 2019.

[20] K. Zhang, D. Lo, E.-P. Lim, and P. K. Prasetyo, "Mining indirect antagonistic communities from social interactions," *Knowledge and Information Systems*, vol. 35, no. 3, pp. 553–583, 2013.

[21] A. Patidar, V. Agarwal, and K. Bharadwaj, "Predicting friends and foes in signed networks using inductive inference and social balance theory," in *2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pp. 384–388, Istanbul, Turkey, 2012.

[22] K. K. Bharadwaj and M. Y. H. Al-Shamri, "Fuzzy computational models for trust and reputation systems," *Electronic Commerce Research and Applications*, vol. 8, no. 1, pp. 37–47, 2009.

[23] V. Kant and K. K. Bharadwaj, "Fuzzy computational models of trust and distrust for enhanced recommendations," *International Journal of Intelligent Systems*, vol. 28, no. 4, pp. 332–365, 2013.

[24] C. Zhou, A. Li, A. Hou et al., "Modeling methodology for early warning of chronic heart failure based on real medical big data," *Expert Systems with Applications*, no. article 113361, 2020.

[25] S. Saha, M. Pal, and A. Konar, "Triangular membership function based real-time gesture monitoring system for physical disorder detection," *Computing and Visualization in Science*, vol. 22, no. 1-4, pp. 1–14, 2019.

[26] Q. He, G. Cui, X. Zhang et al., "A game-theoretical approach for user allocation in edge computing environment," *IEEE Transactions on Parallel and Distributed Systems*, vol. 31, no. 3, pp. 515–529, 2019.

[27] X. Xia, F. Chen, Q. He, J. C. Grundy, M. Abdelrazek, and H. Jin, "Costeffective app data distribution in edge computing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 32, no. 1, pp. 31–44, 2020.

[28] Z. Zhou, Q. J. Wu, Y. Yang, and X. Sun, "Region-level visual consistency verification for large-scale partial-duplicate image search," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 16, no. 2, pp. 1–25, 2020.