

# Link User Identities Across Social Networks Based on Contact Graph and User Social Behavior

ZHANGFENG YIN<sup>1</sup>, YANG YANG<sup>1</sup>, AND YUAN FANG

School of Computer Science and Information Engineering, Hubei University, Wuhan 430062, China

Corresponding author: Yang Yang (yangyang@hubu.edu.cn)

This work was supported by the National Natural Science Foundation of China under Grant 62002104.

**ABSTRACT** With the rapid development of Social Networking Services (SNSs), linking online user IDs is becoming increasingly important to internet service providers. Existing methods can achieve matching adjacent IDs between different services, where adjacent IDs mean the IDs that send message loggings at the same physical location. However, nonadjacent IDs also need to be matched in reality, which is a key challenge. In this paper, a new method based on users social behaviors and contact graph is put forward to realize linking of IDs across domains. This method can be used for matching both adjacent IDs and nonadjacent IDs. Specifically, all the IDs are mapped to contact graph. And we utilize a set matching algorithm based on the contact graph to find out the set of candidate IDs and generate confidence score by means of this algorithm to select the most appropriate matching. Our experimental results show that our algorithm is capable of identifying not only the set of adjacent IDs that belong to one same user but also the set of nonadjacent IDs that belong to one same user.

**INDEX TERMS** User identify linkage, contact graph, social networks.

## I. INTRODUCTION

There are various online services which play an important role in our daily life. It is very normal for an ordinary user to have two or more online IDs in different servers. For instance, a user can log in Twitter, Foursquare and Facebook simultaneously [21], [22]. In addition, a user may have several online IDs at the same server, with different IDs playing different roles. As user IDs offer abundant data, service providers are highly motivated to mine customer data and optimize user experience. To comprehend user behaviors more comprehensively, it's increasingly important to link user IDs among different services so as to merge separated data [1], [2].

Therefore, linking online user IDs has a profound influence on service providers. To cater to the development of SNSs, the methods for Linking IDs have developed from the ones of linking IDs on the same platform to those for linking IDs across domains. The methods of linking IDs on the same platform are mainly used for linking user IDs by relying on the user data of specific services, such as user profiles and

The associate editor coordinating the review of this manuscript and approving it for publication was Khin Wee Lai<sup>1</sup>.

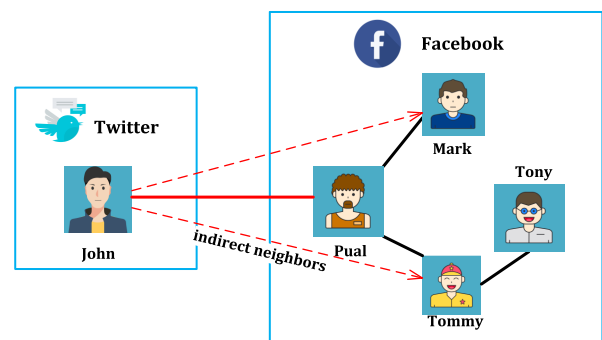
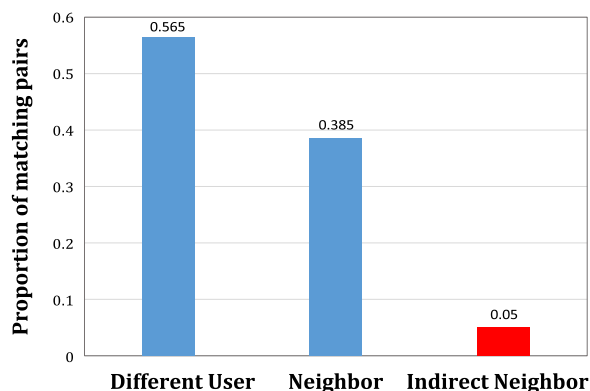


FIGURE 1. Neighbors and indirect neighbors of object ID.

social graph [3], [4]. However, it's difficult to match IDs on more than one server by linking IDs on the same platform. The methods of linking IDs across domains mainly consist of a method based on trajectory data and a method based on contact graph. Specifically, the method based on trajectory data is applicable for matching IDs in a one-to-one manner between different servers [10]. However, this method still

faces key challenge when there are more than two user IDs on the same platform. The method based on contact graph can be used for matching IDs in a one-to-many manner between different servers, in which the set of candidate IDs for object ID is defaulted as the neighbor of object ID [15]. However, we found that non-neighbors of object ID may also belong to one same user as object ID [14], [22], [25]. More exactly, an indirect neighbor of object ID may also belong to one same user as object ID, see Fig. 1. The figure 1 not only shows that John and John's neighbors may belong to one same user, but also shows that John and John's indirect neighbors Mark and Tommy belong to one same user. After statistics on the real data set, we found that matching pairs that are indirectly adjacent to object ID and belong to one same user account for the proportion of the total number of matching pairs as shown in Figure 2 [22]. In Figure 2, Different user indicates the proportion of matching pairs with a matching probability of 0 in the matching result, Neighbor indicates the proportion of matching pairs that are adjacent to object ID and belong to one same user in the matching result, and Indirect neighbor indicates the proportion of matching pairs that are indirectly adjacent to object ID and belong to one same user in the matching result. None of existing methods is capable of capturing IDs that are indirectly adjacent to object ID and belong to one same user as object ID. The goal of this paper is to link indirect neighbors and neighbors which belong to one same user as object ID in different services.



**FIGURE 2.** Proportion of user identity matching pairs in the real data set.

In this paper, a method based on contact graph and user social behaviors is proposed to link IDs, which is capable of identifying not only the set of adjacent IDs that belong to one same user but also the set of nonadjacent IDs that belong to one same user. Firstly, all IDs are mapped to a big graph by using contact graph model, in which nodes are user IDs and two nodes on both ends of one side mean that these two nodes have accessed to the same physical location. Secondly, the goal of this paper is to link some IDs which are indirect neighbors of the object ID, and integrate them with the neighbors of the object ID to form a set of candidate IDs. In order to gain the IDs which are indirect neighbors

of object ID, the indirect neighbors are preprocessed through link prediction in our research [25]. Meanwhile, a universal model for users' social behaviors is built in this paper. Finally, we utilize a set matching algorithm based on the contact graph and the universal model to dispose the set of candidate IDs, and select the most appropriate match in line with confidence score gained by means of this algorithm.

The performance of the algorithm in this study is measured on two real data sets. The first dataset is the Twitter-Foursquare dataset. The second dataset is the Gowalla-Brightkite dataset, which comes from the public website snap. At the same time, this article uses the two most advanced algorithms of SIMP and CN as the baseline algorithm. Our experimental results show that our algorithm is capable of identifying not only the set of adjacent IDs that belong to one same user but also the set of nonadjacent IDs that belong to one same user. In summary, we have made the following contributions:

- (i) A new method based on users social behaviors and contact graph is put forward to realize linking of IDs across domains.
- (ii) Our algorithm is capable of identifying not only the set of adjacent IDs that belong to one same user but also the set of nonadjacent IDs that belong to one same user.

## II. RELATE WORK

The related studies, which contain the methods for linking IDs on the same platform and the method for linking IDs across domains, are introduced in this section.

### A. LINKING IDS ON THE SAME PLATFORM

As the mainstream methods, the methods for linking IDs on the same platform were extensively researched in the early days when the internet was not so widely popularized. Goga *et al.* [3] used user profile attributes (such as user name, profile photo) and certain similarity features (such as posting timestamp and writing style) to link ID, which can be largely reliable in practice match user's identity in practice. Korula *et al.* [4] utilized the social graph method to link user IDs. This is the first time that someone has formalized the user identity linking problem, and designed an effective, partial, and simple parallel algorithm to solve it. Goga *et al.* [5] connects users based on the similarity of the movie ratings of Netflix and IMDB. Zafarani *et al.* [6] proposed the MOBIUS algorithm of behavior modeling to realize link ID, which realizes the mapping of individual identities on media websites. D. Liu *et al.* used the CT method to reduce the energy loss during data transmission [39]. Narayanan *et al.* [7] proposed a new de-anonymity attack method, which performs well in the case of uneven data and lack of background. Mu *et al.* [19] can observe the real identity correspondence in social networks more naturally through the temporal and spatial location of potential users. Nevertheless, these methods which are only applicable to linking IDs on the same platform cannot cater to the development of SNSs.

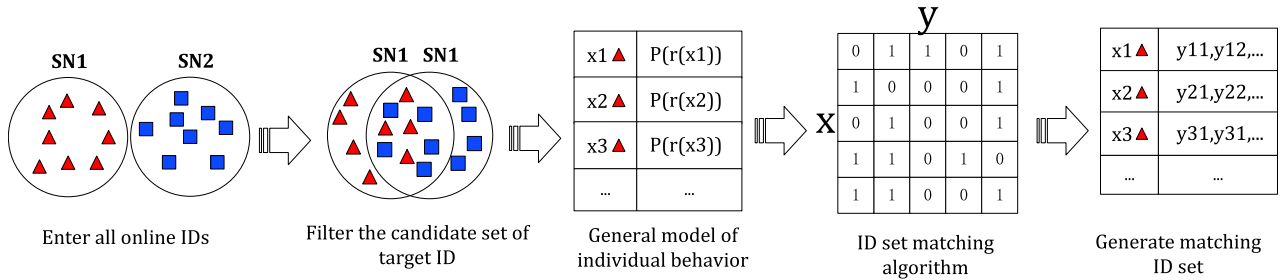


FIGURE 3. The overall flow of our method.

**B. LINKING IDS ACROSS DOMAINS**

The method for linking IDs across domains has a far-reaching influence on service providers. Existing method can be used for matching IDs in a one-to-one and one-to-many manner between different servers. In terms of matching IDs in an one-to-one manner, Riederer and Rossi link IDs on the basis of LBSs [35], [36]. Specifically, Riederer et al. [10] proposed an effective and general method to solve the coordination problem based on location data sets, which use any pair of sporadic location data sets to determine the most likely match. Rossi et al. [11] proposed a linking method based on trajectory data, which can obtain the spatiotemporal data created by the user over time and the frequency of the place visited. At the same time, Jiang Hongbo and others not only applied LBSs technology to mobile computing, but also solved the related problems of indoor navigation [37], [38]. In addition, R.Zafarani et al. [8] links the same users through different platforms so that they can fully understand user behavior and provide better recommendations. Li, C. et al. [24] match user identities between two servers through a mapping relationship. Although these methods can match IDs between two servers in an one-to-one manner, they face great difficulties in terms of ID diversity. In terms of one-to-many ID matching between two servers, X. Han et al. [12] used the location data link ID generated by users in social media platforms and proposed a framework based on copolymerization. Seglem et al. [13] integrated the profile files of different services through the user’s spatio-temporal location information. However, the study did not attempt to preserve user privacy, and was even considered as an attempt to infringe user privacy. Vosecky et al. [9] proposed a user identification method based on web profile matching, and combined with the user’s friend network to further extend the effectiveness of this method, but it also depends on the characteristics of specific services (such as social graphs). Huandong Wang et al. [15] proposed SIMP algorithm based on the connection graph and the temporal and spatial locality of user activities. The SIMP algorithm can ensure that when users access online services at will, they can associate users with actual locations and time. However, the SIMP algorithm cannot capturing IDs that are not adjacent and belong to one same user.

In sum, to adapt to the development of SNSs, the methods for linking user IDs are kept on improving, and have developed from the ones for linking IDs on the same platform to those for linking IDs across domains. In terms of linking IDs across domains, Huandong Wang et al. proposed SIMP algorithm based on the connection graph and the temporal and spatial locality of user activities. SIMP algorithm based on contact graph can be used for matching IDs in a one-to-many manner between different servers, in which the set of candidate IDs for object ID is defaulted as the neighbor of the object ID [15]. However, it is found out that a non-neighbor of the object ID may also belong to one same user as the object ID [14], [22], [25]. This paper proposes a new ID matching algorithm based on Bayesian theory. Our method can identify not only adjacent IDs belonging to one same user, but also non-adjacent IDs belonging to one same user.

**III. METHODOLOGY**

In this section, we first present the overall flow of our method in figure 3. Secondly, some basic concepts and candidate ID sets are introduced. At the same time, a probabilistic model describing the social behavior of users is presented. Finally, we clarify the goal of this paper and propose an algorithm for matching ID.

**A. PROBLEM DEFINITION**

In this study,  $\mathcal{S}$  represents the set of online ID types and  $\mathcal{B}$  denotes the set of online IDs. For each online ID  $x$ , its ID type is  $s(x)$ .  $\forall s \in \mathcal{S}$ ,  $\mathcal{B}^s$  represents the set of all ids of type  $s$ .

For an arbitrary online ID  $y \in \mathcal{B}$ , its mobile record is expressed as  $r(y) = \{(l_1, t_1), (l_2, t_2), \dots\}$ , where  $(l_s, t_s)$  represent a mobile record at location  $l_s$  and time  $t_s$ . In addition, the set of all time bins is denoted as  $\mathcal{T}$ , and the set of all regions is denoted as  $\mathcal{L}$ .

Furthermore, the movement record of an ID set  $\delta$  is defined as  $r(\delta) = \{r(z)|z \in \delta\}$ . For each pair of online ID  $y, x \in \mathcal{B}$ , whether the two IDs belong to one same user is represented as a binary variable  $SU(y, x)$ . That is,

$$SU(y, x) = \begin{cases} 1, & \text{if } y, x \text{ belong to one same user} \\ 0, & \text{otherwise} \end{cases}$$

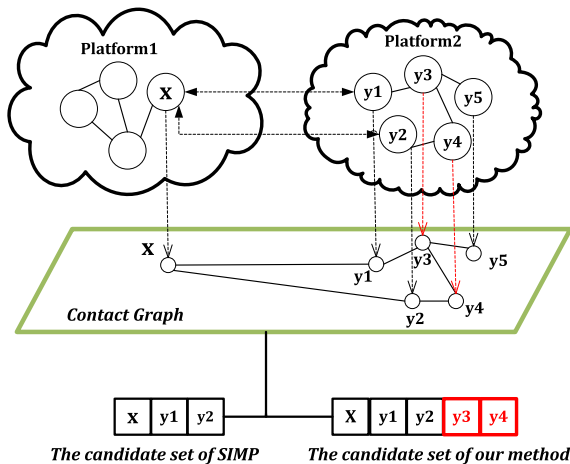
We can also apply this to a more general situation. For an online ID set  $\delta$ , whether they belong to the same user is represented as variable  $SU(\delta) = \prod_{y,x \in \delta} SU(y, x)$ . In a similar fashion, for an online ID  $x$  and a set of IDs  $\delta$ , whether they belong to the same user is represented as variable  $SU(\delta, x) = \prod_{y \in \delta} SU(y, x)$ .

**Definition 1 (Contact Graph):** The contact graph of IDs is expressed as  $G = (B, E)$ . For a pair of online IDs  $x, y \in B$ , if  $x$  and  $y$  have posted message records at the same location, then there is an edge between  $x$  and  $y$  in  $E$ , i.e.,  $\exists l \in L$ , such that  $(l, t_1) \in r(x)$  and  $(l, t_2) \in r(y)$  hold for some  $t_1, t_2 \in T$ .

For two online IDs  $x, y \in B$ , if  $x$  and  $y$  have posted message records at the same location, then  $x$  is the neighbor of  $y$ . For three online IDs  $x, y, z \in B$ , if  $x$  is the neighbor of  $y$ ,  $y$  is the neighbor of  $z$ , then  $x$  is the indirect neighbor of  $z$ . As shown in Figure 1, John is not only Paul’s neighbor, but also an indirect neighbor of Mark and Tommy.

**B. FILTER THE CANDIDATE SET OF OBJECT ID**

Considering the candidate set of object ID is based on contact graph, it is of great importance for us to understand the implications of the graph. In order to integrate the information of different services, all of the IDs that have different relationships with each other are mapped to the graph. In view of the method for linking IDs across domains, Huandong Wang *et al.* believed that the candidate set of object ID is limited to its neighbors, and thus proposed a SIMP algorithm. As shown in Fig. 4, the candidate set of object ID  $x$  is  $\{y_1, y_2\}$ . However, enlightened by Wei Chen *et al.* [22], we found that the object ID and its indirect neighbors might belong to one same user, and at the same time, we did identify such a circumstance in real data sets. Therefore, we consider the candidate set of object ID as the neighbors and indirect neighbors of object ID. As shown in Fig. 4, the candidate set of object ID  $x$  is  $\{y_1, y_2, y_3, y_4\}$ .



**FIGURE 4.** The figure shows the object ID candidate set of the SIMP algorithm and the object ID candidate set of our method.

For any  $x \in B$ , the candidate set of object ID  $x$  is divided into two parts: neighbor  $\mathcal{N}(x) = \{b \mid b \in B, (b, x) \in E\}$

of the object ID and indirect neighbor  $\mathcal{I}(x)$  of the object ID. To obtain more accurate results, the neighbor of object ID is cut into  $\mathcal{N}^s(x) = \mathcal{N}(x) \cap B^s$ . Also, we cut its indirect neighbor into  $\mathcal{I}_q^s(x) = \mathcal{I}_q(x) \cap B^s$  in combination with the link prediction method [25]. Thus, our candidate ID set is  $C(x) = \mathcal{N}^s(x) \cap \mathcal{I}_q^s(x)$ .

**C. GENERAL MODEL OF INDIVIDUAL BEHAVIOR**

To clearly describe the problem for linking IDs across domains, this study constructed a general model based on user behavior. Established by the spatio-temporal localization of user activities. This model may be used to describe how users generate records at different locations.

On the one hand, in order to model various behaviors of users when accessing the server, we binarized the number of records in the discrete time bin, and simplified it based on this [10]. Meanwhile, this study considered that the user’s access to position  $l$  at each discrete time bin followed the Bernoulli distribution with probability  $q_l$ . On the other hand, when the user visits location  $l$ , whether there is a record with ID type  $s$  follows Bernoulli distribution with probability  $q_s$ . To sum up, if ID  $x \in B$ , the probability generated by the observation record is as follows:

$$P(r(x)) = \prod_{l \in L} \prod_{t \in T} \left[ (1 - q_l)^{1 - F_x(l,t)} + q_l q_{s(x)}^{F_x(l,t)} (1 - q_{s(x)})^{1 - F_x(l,t)} \right],$$

where  $F_x(l, t)$  is the judgment function of whether the mobile record  $(l, t)$  exists in  $r(x)$ .

We can estimate  $q_l$  and  $q_s$  from the direction of probability theory. In addition, the number of records accessed by ID  $x$  in server  $s$  and location  $l$  is expressed as  $N_l^s$ . Therefore, the total number of records of all users accessing server  $s$  is expressed as,

$$|B^s| \cdot q_s \cdot |T| = \sum_{l \in L} N_l^s \tag{1}$$

For any position  $l$ , the number of expected records is expressed as:

$$|B^s| \cdot q_l \cdot q_s \cdot |T| = N_l^s \tag{2}$$

By combining equations (1) and (2), we achieve the estimation for  $q_l$  and  $q_s$ .

Furthermore, for a set of online id  $\delta \subseteq B$ , their observation records are generated with the following probability under the condition that they belong to the same user:

$$P(r(\delta) \mid SU(\delta) = 1) = \prod_{l \in L, t \in T} \left[ (1 - q_l)^{1 - F_\delta(l,t)} + q_l \prod_{w \in \delta} q_{s(w)}^{F_w(l,t)} (1 - q_{s(w)})^{1 - F_w(l,t)} \right],$$

where  $F_\delta(l, t)$  is the judgment function of whether the mobile record  $(l, t)$  exists in  $r(\delta)$ .

**D. ID SET MATCHING PROBLEM**

For any object ID  $x \in \mathbf{B}$ , our goal is to find a set of online IDs that belong to one user as  $x$ . At the same time, our goal is formally defined as: *User Identify Set Matching Problem*(UISMP).

*Given:* Object ID  $x$  and its movement record  $r(x)$ . The candidate set of object ID  $\delta_1, \dots, \delta_N \subseteq \mathbf{B}$  and their movement records  $r(\delta_i)$ , where  $i = 1, \dots, N$ .

*Problem:* We must get a ranking function  $\phi : \{\delta_1, \dots, \delta_N\} \rightarrow \{1, \dots, N\}$ . This ranking function can make the IDs belonging to one same user as  $x$  arranged as high as possible, and the ranking function is expressed as:

$$\min_{\phi} \sum_{i=1}^N SU(\delta_i, x) \phi(\delta_i) \tag{3}$$

The goal of our algorithm is to get the ranking function  $\phi(\delta_k)$  of each candidate ID set  $\delta_k$ . To be specific, based on the movement records of the same user in  $\delta_k$ , we get a joint probability  $P(SU(\delta_k, x) = 1 | r(\mathbf{W}))$ , where  $\mathbf{W}$  is the set of candidate IDs. At the same time, the candidate ID set is ranked on the basis of this joint probability. In addition,  $\mathbf{W}$  is set to  $C(x)$ . We describe in detail how to calculate  $P(SU(\delta_k, x) = 1 | r(\mathbf{W}))$  later.

We first consider the one-to-one matching problem, which is to link the ID pairs belonging to the same user in the two services. In the case of pairwise matching, each user can only have at most one user in each service. At the same time, assuming that the server of the object ID  $x$  is  $s_0$  and the server of the candidate ID is  $s_1$ , we get  $\mathbf{W} = C^{s_1}(x) \cup x$ , as shown in Figure 1. In more depth, according to the probabilistic model of user social behavior in (3.3), we obtain the probability  $P(SU(y, x) = 1 | r(\mathbf{W}))$  that  $y$  and  $x$  belong to one same user, which solves the one-to-one matching problem [15].

So far, the one-to-one matching problem has been solved. However, under normal circumstances, there may be multiple IDs in  $C^{s_1}(x)$  that belong to one same user as the object ID  $x$ , as shown in Figure 2. On the basis of the one-to-one matching problem, this section further studies the one-to-many matching problem.

When we consider multiple ID matching, for example,  $y_1, y_2 \in C(x)$ , variables  $SU(y_1, x)$  and  $SU(y_2, x)$  are not independent of each other. This means that we cannot directly use the product of each ID probability as the joint probability. For an ID set  $\delta \subseteq \mathbf{W}$ , to calculate  $P(SU(\delta, x) = 1 | r(\mathbf{W}))$ , we should first obtain:

$$\begin{aligned} P(SU(\delta, x) = 1 | r(\mathbf{W})) &= P(r(\mathbf{W}), SU(\delta, x) = 1) / P(r(\mathbf{W})), \\ &\propto P(r(\mathbf{W}), SU(\delta, x) = 1) \end{aligned}$$

At the same time, the previous equations are dealt with more deeply. Simplifying the above formula through the total probability formula of all partitions of  $\mathbf{W}$ , we acquire:

$$P(r(\mathbf{W}), SU(\delta, x) = 1)$$

$$= \sum_{h \in \mathcal{P}(\mathbf{W})} P(r(\mathbf{W}), SU(\delta, x) = 1 | h)P(h)$$

$P(h)$  represents the prior probability of partition  $h$ , and the concept of partition is mentioned in H. Wang’s research [20]. More precisely, if  $\delta$  and  $x$  are divided into a set  $h$ , then  $P(SU(\delta, x) = 1 | h) = 1$ ; otherwise,  $P(SU(\delta, x) = 1 | h) = 0$ . Therefore, we first get all the IDs within the set  $\delta \cup x$  and then denote the set of all partitions as  $\mathcal{P}(\mathbf{B}, \delta \cup x)$ . Based on the Bayesian theory, we can obtain equation (4) by combining the relationship between  $P(r(\mathbf{W}), SU(\delta, x) = 1)$  and  $P(SU(\delta, x) = 1 | r(\mathbf{W}))$ ,

$$P(SU(\delta, x) = 1 | r(\mathbf{W})) \propto \sum_{h \in \mathcal{P}(\mathbf{W}, \delta \cup x)} P(r(\mathbf{W}) | h)P(h) \tag{4}$$

Besides, for any partition  $h \in \mathcal{P}(\mathbf{W})$ , we use  $M(h)$  to approximate  $P(r(\mathbf{W}) | h)P(h)$ , which is expressed as follows:

$$M(h) = P(r(\mathbf{W}) | h)P(h) = P(h) \prod_{\lambda \in h} P(r(\lambda) | SU(\lambda) = 1)$$

Putting it into (4), we obtain:

$$P(SU(\delta, x) = 1 | r(\mathbf{W})) \propto \sum_{h \in \mathcal{P}(\mathbf{W}, \delta \cup x)} M(h) \tag{5}$$

There is another contrasting situation that we need to consider  $SU(\delta, x) \neq 1$  corresponds to certain other partitions, which are the partitions in  $\delta \cup x$  that are not divided into a set. Therefore, we also obtain:

$$P(SU(\delta, x) \neq 1 | r(\mathbf{W})) \propto \sum_{h \in \mathcal{P}(\mathbf{W}) \setminus \mathcal{P}(\mathbf{W}, \delta \cup x)} M(h) \tag{6}$$

By combining (5) and (6), we have:

$$\begin{aligned} P(SU(\delta, x) = 1 | r(\mathbf{W})) &= \sum_{h \in \mathcal{P}(\mathbf{W}, \delta \cup x)} M(h) / \sum_{h \in \mathcal{P}(\mathbf{W})} M(h) \end{aligned} \tag{7}$$

So far, this paper has obtained the probability of ID matching through a series of derivations, which solves UISMP problem.

**E. SIMILARITY ALGORITHM**

According to the introduction in section D, we have solved the UISMP problem. However, there is still a problem with calculations. The problem is that the computational complexity of formula equation (7) is high, which leads to much calculation. To solve the above problems, we adopt the following three methods:

- Ignore some non-adjacent ids: If two IDs are neither adjacent nor indirectly adjacent, then the two IDs do not belong to one same user. Therefore, the candidate ID set of object ID  $x$  is limited to the neighbors and indirect neighbors, and this candidate ID set can be expressed as  $C(x)$ . As a result, we can greatly reduce the size of  $\mathbf{W}$  and further reduce them.
- Ignore the denominator: Because the denominator in the result of equation (7) has nothing to do with  $\delta$ , we only

**Algorithm 1**  $RS(x, C(x), \delta)$ 

**Input:** The object ID  $x$ , its candidate set  $C(x)$  and a set of IDs  $\delta \subseteq C(x)$ ;

**Output:** Ranking score based on probability  $RS(x, C(x), \delta) = P(SU(\delta_1, x) = 1 | r(\mathbf{W}))$ ;

```

1: Initialize:  $M_{sum} \leftarrow 0; M_{target} \leftarrow 0; f \leftarrow 0$ ;
2: if  $|C(x)| > N_{max}$  then
3:    $f = 1$ ;
4: end if
5: if  $f = 0$  then
6:   for  $h \in \mathcal{P}(C(x))$  do
7:      $M_{sum} = M_{sum} + M(h)$ ;
8:     if  $\exists U \in h$  s.t.  $\delta \cup x \subseteq U$  then
9:        $M_{object} = M_{object} + M(h)$ ;
10:    end if
11:  end for
12:   $CP(\delta, x) = M_{object} / M_{sum}$ ;
13: else
14:    $h = \{\delta \cup x\} \cup \{\{w\} | w \in \mathbf{W} \setminus \{x \cup \delta\}\}$ ;
15:    $CP(\delta, x) = M(h)$ ;
16: end if

```

sort the set of candidate IDs based only on the numerator of equation (7). Therefore, we only need to get the data in  $\mathcal{P}(\mathbf{B}, \delta \cup x)$ .

- Reduce feasible partitions: In order to reduce the computational complexity, we adjust the feasible region to  $\mathcal{P}(\mathbf{B}, \delta \cup x)$ . Specifically,  $|\mathbf{W}|$  is reduced to  $N_{max}$ . If  $|\mathbf{W}| \geq N_{max}$ , all IDs in  $\mathbf{W} \setminus \{x \cup \delta\}$  belonging to different users.

Based on the similarity method and equation (7), we propose an algorithm for calculating the confidence level, as described in Algorithm 1. After three approximation methods, the complexity of our algorithm is reduced. Given the object ID  $x$  and its candidate set  $C(x)$ , if  $|C(x)| \leq N_{max}$ , then we can calculate the confidence score by traversing all the partitions in  $\mathcal{P}(\mathbf{W})$  according to equation (7). Otherwise, we only use two methods to reduce computational complexity.

#### IV. PERFORMANCE EVALUATION

First, this chapter briefly introduces the two original data sets used in the experiment. The two data sets are mainly the Gowalla-Brightkite data set and the Twitter-Foursquare data set. Second, this article explains the content of the two baseline algorithms. Third, the evaluation indicators of the experiment and the results of the experiment analysis are supplemented.

##### A. DATASETS

The performance of the algorithm is evaluated on two real data sets, which come from papers by SNAP and Huandong Wang respectively.

##### 1) GOWALLA-BRIGHTKITE

Users have relevant dynamic information on the Gowalla and Brightkite platforms. Therefore, we get the real mapping between Brightkite account and Gowalla account. This dataset can be retrieved on SNAP. Simultaneously, the dataset has a total of 58,228 users and 441,143 check-in locations. However, only part of the original data set was used during the experiment, namely, their movement trajectories [27], [30], [31], [34].

##### 2) TWITTER-FOURSQUARE

As we all know, Foursquare and Twitter are two social networks with a large number of users worldwide. It is worth noting that users can publish data related to time and space on these two social network platforms. To evaluate the performance of our algorithm, our model is trained on Foursquare and Twitter data sets. However, only part of the original data set is used in training, namely their movement trajectories [16], [26], [32].

#### B. BASELINE ALGORITHMS

The two baseline algorithms are elaborated as follows:

##### 1) SIMP

In order to more accurately describe the daily behavior of users, Huandong Wang *et al.* established a connection graph model, which maps all accounts to a large graph. At the same time, the SIMP algorithm was proposed on the basis of the contact graph model. In order to prove the optimality of the algorithm, Huandong Wang used a Bayes-based method to calculate the confidence probability of the algorithm. Finally, the algorithm solves the problems of inconsistent data quality and ID diversity when linking across services [15], [16].

##### 2) COMMON NEIGHBOR (CN)

The common neighbor algorithm proposed by Dashun Wang and others can effectively solve some link problems [17], [18]. Specifically, they use the number of common neighbors between two nodes to determine the similarity between the two nodes. Therefore, we regard each ID as a point, the IDs that have visited the message record in the same physical location as neighbors, and measure their similarity by the number of public access records between the two IDs.

#### C. EVALUATION METRICS

We choose three evaluation indicators, including recall, precise and AUC. These three standard indicators are used to evaluate our system performance. More specifically, we use an algorithm to generate a set list for each target ID:  $[x_1, x_2, \dots, x_k]$ , where  $x_i$  represents the  $i_{th}$  ID, and  $k$  represents the number of matching IDs. Secondly, we use the algorithm to calculate the three performance indicators of the list. The detailed calculation process is as follows:

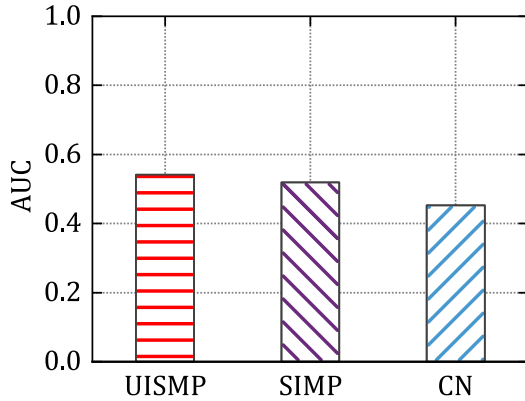


FIGURE 5. AUC for one-to-many relationships(Twitter vs. Foursquare).

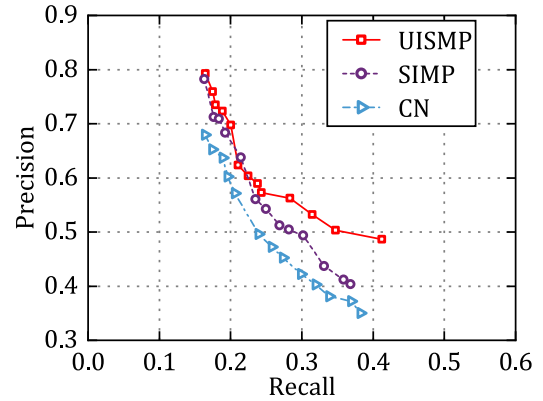


FIGURE 6. Precision and Recall in one-to-many relationships (Twitter vs. Foursquare).

1) ID LIST EVALUATION

After setting the set size to 1 ( $|Y_i| = 1$ ), our set list is transformed into ID list. Therefore, for each set of set lists  $[Y_1, Y_2, \dots, Y_k]$ , we combine the highest-ranked IDs in each set into a new ID list, and use AUC, precision, and recall to evaluate the list.

*Precision & Recall* : Precision is defined as the proportion of user pairs that return the correct link contained in the result. Recall is defined as the proportion of actual linked user pairs included in the returned results [29], [33].

$$Recall = \frac{\alpha}{\beta}, Precision = \frac{\alpha}{\gamma}$$

Among them,  $\gamma$  is the number of ID pairs that are actually linked in the real data,  $\beta$  is the number of ID pairs that are returned, and  $\alpha$  is the number of ID pairs that are actually linked in the returned result.

*AUC*: AUC in machine learning books means the area under the ROC curve [23], [28]. The curve here represents the relationship between the true positive rate (TPR) and the false positive rate (FPR). The value of AUC is higher than the probability of choosing a positive instance to choose a negative instance. This value is mainly the evaluation value of the accuracy of the permutation function, namely:

$$AUC = \frac{\sum_{i=1}^{m_0} (m_0 + m_1 - r_i) - m_0(m_0 + 1) / 2}{m_0 m_1},$$

where  $m_0$  and  $m_1$  respectively represent the number of positive and negative instances, and  $r_i$  represents the rank of the  $i_{th}$  positive instance. Among them, the positive instance indicates that it is correct on the basis of real data. We set the  $K$  value of the ID list to  $k = 10$ , so  $m_0 + m_1 = k = 10$ .

D. EXPERIMENT AND RESULTS

We conduct three sets of experiments on Twitter-Foursquare and Gowalla-Brightkite. As can be seen from figure 5, figure 6, figure 7 and figure 8, we evaluate our algorithm based on the conclusion of the comparison. AUC is used to evaluate our system. Simultaneously, precise and recall are used to evaluate the accuracy of our system.

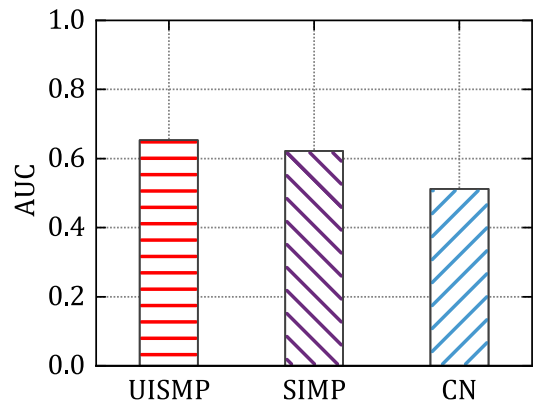


FIGURE 7. AUC for one-to-many relationships (Gowalla vs. Brightkite).

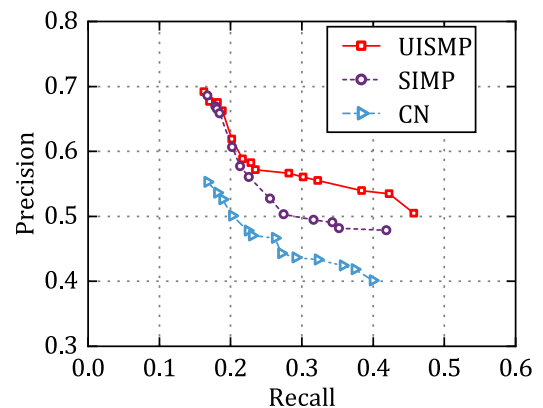


FIGURE 8. Precision and Recall in one-to-many relationships (Gowalla vs. Brightkite).

First, we use UISMP, SIMP and CN to do experiments on the Twitter-Foursquare platform. The experimental results are shown in Figure 5 and Figure 6. Furthermore, the AUC corresponding to UISMP is slightly better than other algorithms in the results, as shown in Figure 5. At the same time, Precise corresponding to UISMP is slightly higher than other algorithms in the results, as shown in Figure 6.

Second, we used UISMP, SIMP and CN to do experiments on the Gowalla-Brightkite platform. The experimental results are shown in Figure 7 and Figure 8. Furthermore, the AUC corresponding to UISMP is slightly better than other algorithms in the results, as shown in Figure 7. At the same time, Precise corresponding to UISMP is slightly higher than other algorithms in the result, as shown in Figure 8.

Thirdly, through the comparison of these groups of experiments, we found that our algorithm UISMP is superior to the other two baseline algorithms in terms of AUC. In addition, when the recall is the same, the precision of our algorithm SIMP is generally higher than that of the baseline algorithm.

## V. CONCLUSION

In this paper, a method based on contact graph and user social behaviors is proposed to link IDs, which is capable of identifying not only the set of adjacent IDs that belong to one same user but also the set of nonadjacent IDs that belong to one same user. Firstly, all IDs are mapped to a big graph by using contact graph model, in which nodes are user IDs and two nodes on both ends of one side mean that these two nodes have accessed to the same physical location. Secondly, the goal of this paper is to link some IDs which are indirect neighbors of the object ID, and integrate them with the neighbors of the object ID to form a set of candidate IDs. In order to gain the IDs which are indirect neighbors of object ID, the indirect neighbors are preprocessed through link prediction in our research. Meanwhile, a universal model for users' social behaviors is built in this paper. Finally, we utilize a set matching algorithm based on the contact graph and the universal model to dispose the set of candidate IDs, and select the most appropriate match in line with confidence score gained by means of this algorithm. This algorithm is capable of identifying not only the set of adjacent IDs that belong to one same user but also the set of nonadjacent IDs that belong to one same user.

## REFERENCES

- [1] M. Yan, J. Sang, T. Mei, and C. Xu, "Friend transfer: Cold-start friend recommendation with cross-platform transfer learning of social knowledge," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2013, pp. 1–6.
- [2] R. Zafarani and H. Liu, "Finding friends on a new site using minimum information," in *Proc. SIAM Int. Conf. Data Mining*, Apr. 2014, pp. 947–955.
- [3] O. Goga, P. Loiseau, R. Sommer, R. Teixeira, and K. P. Gummadi, "On the reliability of profile matching across large online social networks," in *Proc. 21st ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2015, pp. 180–1799.
- [4] N. Korula and S. Lattanzi, "An efficient reconciliation algorithm for social networks," *Proc. VLDB Endowment*, vol. 7, no. 5, pp. 377–388, Jan. 2014.
- [5] 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 (WWW)*, 2013, pp. 447–458.
- [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*, Aug. 2013, pp. 41–49.
- [7] A. Narayanan and V. Shmatikov, "Robust de-anonymization of large sparse datasets," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2008, pp. 111–125.
- [8] R. Zafarani and H. Liu, "Connecting corresponding identities across communities," in *Proc. 3rd Int. AAAI Conf. Web Social Media*, Mar. 2009, pp. 354–357.
- [9] J. Vosecky, D. Hong, and V. Y. Shen, "User identification across social networks using the web profile and friend network," *Int. J. Web Appl.*, vol. 2, no. 1, pp. 23–34, 2010.
- [10] C. Riederer, Y. Kim, A. Chaintreau, N. Korula, and S. Lattanzi, "Linking users across domains with location data: Theory and validation," in *Proc. 25th Int. Conf. World Wide Web*, Apr. 2016, pp. 707–719.
- [11] L. Rossi and M. Musolesi, "It's the way you check-in: Identifying users in location-based social networks," in *Proc. 2nd ACM Conf. Online Social Netw.*, Oct. 2014, pp. 215–226.
- [12] X. Han, L. Wang, L. Xu, and S. Zhang, "Social media account linkage using user-generated geo-location data," in *Proc. IEEE Conf. Intell. Secur. Informat. (ISI)*, Sep. 2016, pp. 157–162.
- [13] E. Seglem, A. Züfle, J. Stutzki, F. Borutta, E. Faerman, and M. Schubert, "On privacy in spatio-temporal data: User identification using microblog data," in *Proc. SSTD*, Cham, Switzerland: Springer, Aug. 2017, pp. 43–61.
- [14] M. Lichman and P. Smyth, "Modeling human location data with mixtures of kernel densities," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2014, pp. 35–44.
- [15] H. Wang, Y. Li, G. Wang, and D. Jin, "You are how you move: Linking multiple user identities from massive mobility traces," in *Proc. SIAM Int. Conf. Data Mining*. Philadelphia, PA, USA: SIAM, May 2018, pp. 189–197.
- [16] J. Zhang, X. Kong, and P. S. Yu, "Transferring heterogeneous links across location-based social networks," in *Proc. 7th ACM Int. Conf. Web Search Data Mining*, Feb. 2014, pp. 303–312.
- [17] S. Daminelli, J. M. Thomas, C. Durán, and C. Vittorio Cannistraci, "Common neighbours and the local-community-paradigm for topological link prediction in bipartite networks," *New J. Phys.*, vol. 17, no. 11, Nov. 2015, Art. no. 113037.
- [18] D. Wang, D. Pedreschi, C. Song, F. Giannotti, and A.-L. Barabasi, "Human mobility, social ties, and link prediction," in *Proc. 17th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2011, pp. 1100–1108.
- [19] X. Mu, F. Zhu, E.-P. Lim, J. Xiao, J. Wang, and Z.-H. Zhou, "User identity linkage by latent user space modelling," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 1775–1784.
- [20] H. Wang, C. Gao, Y. Li, Z. L. Zhang, and D. Jin, "From fingerprint to footprint: Revealing physical world privacy leakage by cyberspace cookie logs," in *Proc. ACM on Conf. Inf. Knowl. Manage.*, Nov. 2017, pp. 1209–1218.
- [21] J. Feng, M. Zhang, H. Wang, Z. Yang, C. Zhang, Y. Li, and D. Jin, "DPLink: User identity linkage via deep neural network from heterogeneous mobility data," in *Proc. World Wide Web Conf. (WWW)*, 2019, pp. 459–469.
- [22] W. Chen, H. Yin, W. Wang, L. Zhao, and X. Zhou, "Effective and efficient user account linkage across location based social networks," in *Proc. IEEE 34th Int. Conf. Data Eng. (ICDE)*, Apr. 2018, pp. 1085–1096.
- [23] K. Shu, S. Wang, J. Tang, R. Zafarani, and H. Liu, "User identity linkage across online social networks: A review," *ACM SIGKDD Explor. Newslett.*, vol. 18, no. 2, pp. 5–17, 2017.
- [24] C. Li, S. Wang, P. S. Yu, L. Zheng, X. Zhang, Z. Li, and Y. Liang, "Distribution distance minimization for unsupervised user identity linkage," in *Proc. 27th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2018, pp. 447–456.
- [25] N. Gupta and A. Singh, "A novel strategy for link prediction in social networks," in *Proc. CoNEXT Student Workshop*, 2014, pp. 12–14.
- [26] R. Kaushal, V. Ghose, and P. Kumaraguru, "Methods for user profiling across social networks," in *Proc. IEEE Int. Conf. Parallel Distrib. Process. Appl., Big Data Cloud Comput., Sustain. Comput. Commun., Social Comput. Netw. (ISPA/BDCloud/SocialCom/SustainCom)*, Dec. 2019, pp. 1572–1579.
- [27] W. Chen, H. Yin, W. Wang, L. Zhao, W. Hua, and X. Zhou, "Exploiting spatio-temporal user behaviors for user linkage," in *Proc. ACM Conf. Inf. Knowl. Manage.*, Nov. 2017, pp. 517–526.
- [28] D. J. Hand and R. J. Till, "A simple generalisation of the area under the ROC curve for multiple class classification problems," *Mach. Learn.*, vol. 45, no. 2, pp. 171–186, 2001.
- [29] D. M. Powers, "Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation," *J. Mach. Learn. Technol.*, vol. 2, no. 1, pp. 37–63, Dec. 2011.
- [30] N. Al Hasan Haldar, J. Li, M. Reynolds, T. Sellis, and J. X. Yu, "Location prediction in large-scale social networks: An in-depth benchmarking study," *VLDB J.*, vol. 28, no. 5, pp. 623–648, Oct. 2019.



- [31] Z. Huo, X. Meng, and R. Zhang, "Feel free to check-in: Privacy alert against hidden location inference attacks in GeoSNs," in *Proc. Int. Conf. Database Syst. Adv. Appl. (ADSA)*. Berlin, Germany: Springer, Apr. 2013, pp. 377–391.
- [32] Y. Shen, T. N. Dinh, H. Zhang, and M. T. Thai, "Interest-matching information propagation in multiple online social networks," in *Proc. 21st ACM Int. Conf. Inf. Knowl. Manage. (CIKM)*, 2012, pp. 1824–1828.
- [33] D. Hand and P. Christen, "A note on using the F-measure for evaluating record linkage algorithms," *Statist. Comput.*, vol. 28, no. 3, pp. 539–547, May 2018.
- [34] Q. Gao, F. Zhou, K. Zhang, G. Trajcevski, X. Luo, and F. Zhang, "Identifying human mobility via trajectory embeddings," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, Aug. 2017, pp. 1689–1695.
- [35] H. Jiang, J. Li, P. Zeng, F. Zeng, Z. Xiao, and A. Iyengar, "Location privacy-preserving mechanisms in location-based services: A comprehensive survey," *ACM Comput. Surv.*, vol. 54, no. 1, pp. 1–36, 2021.
- [36] H. Jiang, M. Wang, P. Zhao, Z. Xiao, and S. Dustdar, "A utility-aware general framework with quantifiable privacy preservation for destination prediction in LBSs," *IEEE/ACM Trans. Netw.*, vol. 29, no. 5, pp. 2228–2241, Oct. 2021.
- [37] Z. Xiao, X. Dai, H. Jiang, and D. Wang, "Vehicular task offloading via heat-aware MEC cooperation using game-theoretic method," *IEEE Internet Things J.*, vol. 7, no. 3, pp. 2038–2052, Mar. 2020.
- [38] H. Jiang, W. Liu, G. Jiang, Y. Jia, X. Liu, Z. Lui, X. Liao, J. Xing, and D. Liu, "Fly-Navi: A novel indoor navigation system with on-the-fly map generation," *IEEE Trans. Mobile Comput.*, vol. 20, no. 9, pp. 2820–2834, Sep. 2021.
- [39] D. Liu, Z. Cao, M. Hou, H. Rong, and H. Jiang, "Pushing the limits of transmission concurrency for low power wireless networks," *ACM Trans. Sensor Netw.*, vol. 16, no. 4, pp. 1–29, Nov. 2020.



**ZHANGFENG YIN** received the B.E. degree from the Hubei University of Automotive Technology, China, in 2015. He is currently pursuing the M.S. degree in electronic and communication engineering with Hubei University, China. His research interests include user identify linkage, computer networks, and machine learning.



**YANG YANG** received the B.E. and M.S. degrees from the Wuhan University of Technology, China, in 2009 and 2012, respectively, and the Ph.D. degree from the Huazhong University of Science and Technology, China, in 2017. He studied with Simon Fraser University, Canada, as a Visiting Student, in 2014. He is currently an Associate Professor with the School of Computer Science and Information Engineering, Hubei University. His research interests include edge computing, mobile computing, and machine learning.



**YUAN FANG** received the B.E. degree from the School of Arts and Sciences, Yangtze University, China, in 2015. He is currently pursuing the M.S. degree in circuit and system with Hubei University, China. His research interests include data mining, computer networks, and machine learning.

...