Balancing User Profile and Social Network Structure for Anchor Link Inferring across Multiple Online Social Networks

Jiangtao Ma, Yaqiong Qiao, Guangwu Hu*, Yongzhong Huang, Meng Wang, Arun Kumar Sangaiah, Chaoqin Zhang and Yanjun Wang

***Abstract*—Along with the popularity of Online Social Network (OSN), more and more OSN users tend to create their accounts in different OSN platforms. Under such circumstances, identifying the same user among different OSNs offers tremendous opportunities for many applications, such as user identification, migration patterns, influence estimation, and expert finding in social media. Different from existing solutions which employ user profile or social network structure alone, in this paper, we proposed a novel joint solution named MapMe, which takes both user profile and social network structure feature into account so that it can adapt more OSNs with more accurate results. MapMe first calculates user similarity via profile features with the Doc2vec method. Then, it evaluates user similarity by analyzing user's ego network features. Finally, the profile features and ego network features were combined to measure the similarity of the users. Consequently, MapMe balances the two similarity factors to achieve goals in different platforms and scenarios. Finally, experiments are conducted on the synthetic and real datasets, proving that MapMe outperforms the existing methods with 10% on average.**

***Index Terms*—Anchor link inferring, graph partition, node matching, online social network, social network analysis**

I. INTRODUCTION

Along with the increasing popularity of online social networks (OSNs), i.e., Facebook, Twitter, Instagram, WeChat, Sina Weibo, OSN user number increases sharply as each user usually has multiple accounts among different OSN platforms. According to the statistics from China Internet Network Information Center (CNNIC) [1], in mainland China only, there are 656 million netizens using smartphones or smart terminal devices to access the Internet by the first half of the year 2016. The top 3 visited OSNs are: WeChat friends' circle (78.7%), QQ friends' zone (67.4%) and Sina Weibo (34%). Meanwhile, another research [2] shows that at least 42% online users access two or more OSNs simultaneously. Inspired by this phenomenon, some applications try to analyze user relevancy that exists in heterogeneous OSNs so as to integrate these OSN services together and provide better services. For instance, recommendation systems giving friend recommendations, location-based systems providing location sharing services, and marketing services expanding their advertise effects etc. All of them perform functions by associating users with different OSNs.

As a matter of fact, user association across multiple OSNs is also known as a social link inferring/prediction problem, which is important and valuable in the recent years, since information can be propagated across multiple networks through user correlation in heterogeneous networks. Currently, more and more applications, i.e., link transferring [3], community detection [4], viral marketing [5], friend recommendation systems [6] across OSNs, are relying on it, which can be beneficial to understand online user behaviors [7] , learn user's different views, and solve the cold start problem in recommend systems [8] etc.

Traditional solutions for this issue can be categorized into three types: (1) user profile based matching method: analyzing user's profiles to identify the same user in different OSNs. However, as many users fake or hide their profiles for privacy concerns, this kind of method may not achieve a good effect; (2) user behavior based matching method: exploiting user's behavioral patterns, i.e., user activities, writing styles, geo-location timestamps of posts, as features to calculate user similarity in different OSNs. However, such solutions cannot scale to general OSNs as they are aimed at some specific platforms only; (3) user topology based matching method: utilizing user's similar social connections in different OSNs [9] to analyze user's ego network [10] (ego network consists of ego node and its neighbors) and map them. However, this method

This work is supported by the National Nature Science Foundation of China (61402255, 61170292, 61373161, 61672470), National key R&D projects intergovernmental cooperation in science and technology of China (12016YFE0100600, 12016YFE0100300), Henan Province Science and Technology Department Foundation (162102410076). Henan Province Educational Committee Foundation (16A520062, 17A520064), Guangdong Natural Science Foundation (2015A030310492) and Fundamental Research Project of Shenzhen Municipality (JCYJ20160301152145171).

J. Ma, Y. Qiao, Y. Huang, M. Wang, C. Zhang, and Y. Wang are with the State Key Laboratory of Mathematical Engineering and Advanced Computing, Zhengzhou450001, Henan, PRC (email: kitesmile2000@gmail.com; qiaoying qiong@126.com; 18600200718@163.com; mengwangchn@outlook.com; zhangcq@zzuli.edu.cn, 64502147@qq.com;). J. Ma and C. Zhang are also with Zhengzhou University of Light Industry, Zhengzhou 450001, Henan, PRC.

G. Hu is with the School of Computer Science, Shenzhen Institute of Information Technology, 518172 Shenzhen, Guangdong, PRC (corresponding author, e-mail: hugw@sziit.edu.cn).

Arun Kumar Sangaiah is with the School of Computer Science and Engineering, VIT University, Vellore 632014, India (e-mail: arunkumarsangaiah@gmail.com).

needs to convert network structures into matrixes, which may incur large delays in matrix computation and comparison since most OSNs are sparse.

In order to solve this problem with desirable accuracy and performance, in this paper, we propose a novel solution named MapMe, to map users across multiple heterogeneous OSNs by balancing user profiles and social network structures. To achieve that, we not only examine user similarity according to their profiles, but also similarity by analyzing user's ego network. Consequently, we weight the two similarity factors so as to balance their effects and meet our goals. Compared with the existing work, the main contributions in our scheme are:

(1) We propose a novel model that employs both user profiles and user relationship network structures as features before balancing them by setting a smooth factor according to OSN differences.

(2) We employ node's k depth degree, clustering coefficient and eigenvector to analyze user's ego network structure similarity for OSN users, which can greatly improve solution efficiency based on the previously related approaches. To the best of our knowledge, this is the first attempt for such goals.

(3) We evaluate our scheme and compare it with the state-of-the-art solutions on three synthetic datasets and three real network datasets. The experimental results demonstrate our scheme outperforms the existing methods with 10 percentages.

The rest of this paper is organized as follows: Section II summarizes the related work. Section III states and formulates the social link inferring problem. Section IV elaborates our scheme and Section V evaluates the scheme and compares it with some classic proposals. At last, Section VI concludes the paper.

II. RELATED WORK

Since user profiles, user generated contents (user behaviors) and topologies (user's social connections) are the three main components in OSNs, we mainly investigated existing solutions from these three aspects.

A. User Profile based Matching Method

At first, researchers try to take user profile features, i.e., username, location, gender, birthday, interests, position etc., to identify the same user among different OSNs. Tan et al. [11] first state that about 50% users use the same username in different OSNs. Based on this result, Zafarani et al. [12] utilize username to match users by adding or deleting the prefix/postfix for usernames. Further, Peritio et al. [13] estimate the uniqueness for usernames by establishing the Markov chain model. Similarly, Liu et al. [14] propose an unsupervised approach which takes the n-gram model to estimate the uniqueness of usernames. Moreover, Iofciu et al. [15] identify users across OSNs by measuring the distance between user profiles based on their IDs and tags. In addition, Zhang et al. [16] use the Jaro-Winkler [17] method to link user accounts among different OSNs with a language model [18]. They first convert user profiles into a bag of word vectors, and then calculate profile similarity by analyzing vector similarity through the cosine distance. Differently, Malhotra et al. [19] propose an user's digital footprint method for the same goal, in which usernames, nicknames, locations, photos are gathered for calculating user profile similarity. Although these schemes can achieve a good performance, the biggest challenge is authenticity and integrality, because users normally will not provide their authentic and complete profiles for privacy concerns. Therefore, profile-based matching methods cannot achieve a good result when the veracity of profiles is not guaranteed.

B. User Behavior based Matching Method

Considering user behaviors are unique and cannot be impersonated easily compare to user profiles, researchers start to resort to this idea to infer users across OSNs. MOBIUS [20] maybe the first attempt with such ideas by extracting user behavior features (e.g., limitation and nickname typing pattern) and analyzing these features with a supervised method. Similarly, Liu et al. [21] predict the social links through integrating user attributes, generated contents and other social behaviors together. Additionally, timeline is another important feature of user behaviors. For example, many users prefer to synchronize their activities on different OSNs, i.e., publishing the same photos on Twitter and Flickr. It inspires Goga et al. [22] to build a similarity matrix that takes username, photo and location similarities into account to improve social link prediction accuracy. Moreover, Nie et al. [23] propose a dynamic core interest mapping algorithm, which considers both user's topological and topic model based on user generated content and ego network. Nevertheless, these methods are confined to some specific OSNs more or less, which makes them cannot be scalable to general OSNs.

C. User's Topology-based Matching Method

User's social connections or ego network can be considered as the third features for social link inferring. Generally, user's ego network can be transformed as an adjacent matrix, thus analyzing and comparing the matrix similarity would achieve the same effect. Tan et al. [11] utilize hypergraphs and express user social links with a matrix. Besides, an alignment framework is used to lower matrix dimension so as to reduce the computation cost. Similarly, Cui et al. [24] employ a graph matching method to identify email correspondents across OSNs, which combines user profiles and topological information together to find the user relationships between email networks and Facebook. Moreover, Man et al. [25] propose an unsupervised model named PALE. It can embed the social links into network structures and train a mapping model to do the same thing as well. In addition, Feng et al. [26] invent two metrics to measure the similarity between users in different OSNs. Also, Zhang et al. [16] propose a method named COSNET that considers both the local and global consistency of OSNs, in which an energy-based model was invented and the Lagrangian relaxation method was trained for the implementation. Besides that, Zhou et al. [27] utilize the friendship structure and develop an FRUI algorithm, whose time complexity is lower than other methods as it utilizes matched users as input parameter to select candidate matching user pairs. Despite the inferring accuracy is much higher than the other approaches, topology-based matching method is difficult to scale to large OSNs since the computing cost of this method is considerably expensive for sparse OSNs.

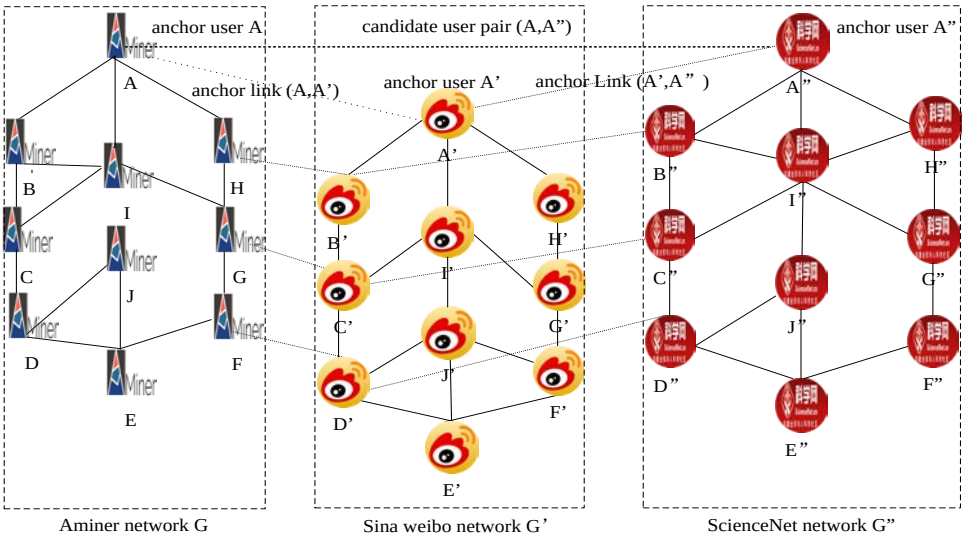


Fig. 1. The illustration of social link inferring problem. There are three OSNs, and each node or anchor, i.e., A, A', A'', in different networks denotes a user account. The thin dashed lines between nodes in different networks (e.g., A-A', A'-A'') are social/anchor links, which indicate the two nodes in the ends of the line belong to the same person. The thick dashed line (A-A") represents candidate user pair. Our goal is to find out the anchor links among different OSNs with desirable performance and accuracy.

Although the above studies give us a lot of inspirations, none of them can satisfy our purpose directly. We will elaborate our solution in detail in the next section.

III. PROBLEM STATEMENT AND FORMULATION

A. Problem Statement

To better understand the problem we are addressing, we take the three typical OSNs, Aminer [1], Sina Weibo [2], and ScienceNet's blog[3] as examples. As Fig. 1 illustrated, Aminer [28] is a semantic-based paper and author indexing system, which provides OSN functions for scientists. Sina Weibo, like Twitter, is a Chinese OSN for users posting and replying messages. According to ScienceNet's blog, it is a popular OSN platform for Chinese researchers. Suppose a user owns an account in each of these networks, e.g., node A, A' and A''. The link between the same user across different networks is called social link or anchor link [29]. In other words, each anchor link refers to a pair of accounts that belong to the same person. So in short, our aim is to accurately and effectively identify social links across multiple OSNs.

However, social link inferring across OSNs has the following challenges: (1) Given mostly neither authentic nor complete user profiles, the first challenge is how to evaluate user profile similarity between different OSNs. (2) Secondly, as most user's ego networks are a large scale of sparse networks, using adjacency matrix to represent this node and its relationship will lead to the curse of dimensionality problem and incur large computation cost as well. (3) Since both profile-based and topology-based solutions have their limitations, the last challenge is how to overcome and integrate them to achieve our goal.

[1] http://www.cn.aminer.org
[2] http://www.weibo.com
[3] http://www.blog.sciencenet.cn

B. Problem Formulation

In order to more accurately portray social link inferring problem, we first give its formulation and assume that the social link indicates a one-to-one relationship. In other words, two edges cannot share the same node. This problem can be converted into a stable matching issue between two OSNs each with one account set and each social link represents a pair of two accounts in different networks belonged to the same user. So our goal can be described as a node matching issue among account sets. Moreover, we can easily extend our proposal to more than two networks.

Definition 1 (social network): Given graph $G = (U, E)$ represents a social network and U represents the set of network user accounts, then E indicates the set of social relationships between users.

Definition 2 (social/anchor link): Given two OSNs G and G', if $(u \in U) \wedge (v' \in U')$, and u and v' belong to the same user, then U and U' are user sets of G and G', respectively. The link between u and v' is called social/anchor link.

Taking Fig.1 illustrated networks as an example, AMINER network, Sina Weibo and ScienceNet's blog are presented by networks G, G', G''. The set of anchor links between G and G' is $((A,A'), (H,B'),(G,C'),(F,D'))$, while the set of anchor links between G' and G'' is $((A',A''),(B',B''),(C',C''),(D',D''))$. Therefore, our objective is find the node pairs between networks, we can describe the objective function as (1):

$$M = Maxnodepair(G, G') \; s.t. \; u \text{ and } v' \text{ is a node pair} \quad (1)$$

Therefore, the anchor link inferring problem is converted into find node pairs between networks with higher precision and recall. Therefore, the anchor link inferring problem is converted into a node mapping problem between two graphs. Thus, we need to find an efficient method to find the mapped node pairs in large social networks. Furthermore, the rule of finding similar nodes across graphs is an important step in anchor link inferring. Consequently,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2017.2717921, IEEE Access

anchor link inferring has the following challenges:

(1) It is challenging to match similar users across large sparse online social networks with an effective and efficient method.

(2) It is challenging to measure the similarity of the user across OSNs with user profile feature and user network structure feature.

(3) It is challenging to define the feature of users and extract features from user profile and network.

IV. SCHEME DETAILS

In this section, we describe our scheme MapMe in detail. First, we divide the social graph into *k* subgraphs with spectral method. Then the subgraphs from two social graphs are matched according to the similarity of subgraphs. After that the nodes pair are mapped between similar subgraphs. In the process of mapping nodes, both profile-based matching and network structure-based matching models are considered. In the former model, profile feature is considered with the Doc2vec [30] method. Therefore, profile similarity is in line with the similarity of Doc2vec converted data vector. While in the network structure based matching model, MapMe utilizes network structure features to measure user similarity. Furthermore, user's *k* depth degree, user's clustering coefficient and User's Eigenvector Centrality are used as features to measure the similarity of the network structure. Eventually, schemes can balance these two similarity factors to achieve our goal.

A. Big Graph Partition

Social network is a big graph with billion nodes, such as Facebook. Therefore, it is difficult to extract features from big graph. Considering the scalability of MapMe, we need to partition the big sparse graph into small subgraphs. To do that, we adopt the spectral method, which is efficient and accurate in sparse graph partition. In detail, MapMe computes the spectrum Λ of adjacency matrix $\boldsymbol{A}$ corresponding to graph G, where λ_i is an eigenvalue, $\Lambda = \{\lambda_1, \lambda_2, \ldots, \lambda_n\}, \lambda_1 \le \lambda_2 \le \ldots \le \lambda_n$. $\boldsymbol{D}$=[d_{ii}] is a degree matrix, d_{ii} is the degree of node i, for Laplace matrix $\boldsymbol{L} = \boldsymbol{D}$-$\boldsymbol{A}$, and all the eigenvalues are non-negative.

After obtaining eigenvalues and eigenvectors of the Laplacian matrix $\boldsymbol{L}$, MapMe can obtain the corresponding graph partition according to the following steps:

(1) Build the Laplacian matrix $\boldsymbol{L}$ for graph G.

(2) Use the spectral method to decompose $\boldsymbol{L}$ and calculate the eigenvalues and eigenvectors.

(3) The feature vector elements are sorted to find the classification point (0, or median, or a value from a more complex classification method). The feature vector is divided into two categories.

(4) Use the top k eigenvectors to partition the graph into k subgraphs, $G_1, G_2, \ldots, G_k$.

After the big graph is divided into k subgraphs, MapMe can extract the features from the subgraphs efficiently. Meanwhile, the feature values of the subgraph, such as node's k depth degree, clustering coefficient and eigenvector centrality can be calculated easily from the subgraphs. In the following section, we will design the method to compute the similarity of two subgraphs.

B. Subgraph Similarity

We utilize spectral method to measure the distance of two subgraphs, and employ this distance measure the similarity of two subgraphs. Since study [31] proves that the graph's structure has a strong relationship with the spectrum of the adjacency matrix, MapMe can utilize this feature to identify OSN users by analyzing the spectrum values in their ego network converted matrixes. Since the isomorphic graphs share the same spectrum, it can distinguish isomorphic graphs from non-isomorphic graphs. As (2) illustrates, MapMe measures two graphic's distance to express the spectrum distance between them, where λ_i and μ_i are the spectrums of graphs $G_1^{'}$ and $G_2^{''}$ represented by the matrix and k is the number of eigenvalues in the matrix. $G_1^{'}$ and $G_2^{''}$ are the subgraphs of G' and G''.

$$d(G_1^{'}, G_2^{''}) = \sqrt{\frac{\sum_{i=1}^{k}(\lambda_i - \mu_i)^2}{\min\{\sum_{i=1}^{k}\lambda_i^2, \sum_{i=1}^{k}\mu_i^2\}}} \tag{2}$$

The similarity of two graphs can be computed as (3):

$$sim(G_1^{'}, G_2^{''}) = \frac{1}{1 + d(G_1^{'}, G_2^{''})} \tag{3}$$

Then we can find the similar subgraphs between G' and G''. After finding the similar subgraphs, we will find node pairs between two subgraphs. In section 4.3, we will propose how to define the similarity of two node pairs. MapMe combines the profile-based method and topology-based method in the following section.

C. Profile-based Similarity

To evaluate user profile based similarity, we utilize the Doc2vec method to represent a document as a feature vector, which can extract features from user profiles. MapMe converts user's profile into a unique vector, which is represented by a column in matrix $\boldsymbol{D}$, and maps every word into a unique vector, represented by a column in another matrix $\boldsymbol{W}$. Therefore, the similarity of the user's profile can be measured through cosine distance of their vectors.

As a result, profile-based document can be embedded into continuous vector space, and thus user similarity can be measured by the cosine distance between two user profile-based vector files, as illustrated in (4).

$$sim_{\text{profile}}(A, B) = \frac{A_{profile} \cdot B_{profile}}{\| A_{profile} \| \cdot \| B_{profile} \|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \sqrt{\sum_{i=1}^{n} B_i^2}} \tag{4}$$

where $A_{profile}$ and $B_{profile}$ mean the vectors of node A and node B, respectively. A_i means the i^{th} attribute of the profile, and n means the amount of profile attributes. From the equation, we can be aware that the higher cosine similarity between vectors, the more similar are the users.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2017.2717921, IEEE Access

D. Network Structure-based Similarity

Besides, we also take social connection based topology structure into account so as to evaluate user network structure based similarity, as illustrated in (5):

$$sim_{structure}(A,B)=\frac{A_{structure}\cdot B_{structure}}{\|A_{structure}\|\cdot\|B_{structure}\|}=\frac{\sum_{i=1}^{n}A_iB_i}{\sqrt{\sum_{i=1}^{n}A_i^2}\sqrt{\sum_{i=1}^{n}B_i^2}} \quad (5)$$

where $A_{structure}=[d_A, ce_A, e_A]$ is a vector that contains user A's ego network features, d_A is the degree of node A, ce_A is clustering coefficient of node A, e_A is the eigenvector of node A. MapMe employs profile-based feature to measure the user's account similarity, and utilizes the network structure feature to measure the user's social relationship similarity. Both of these features are converted into vectors, which can be calculated through cosine similarity. The following subsection will elaborate these network structure details.

1) User's k Depth Degree

Degree Centrality [32] is a measure of node centrality in network analysis. The greater the degree of nodes in a node means that the higher the degree of the center of the node, the more important the node in the network.

The degree distribution of a large-scale real network is subject to a power-law distribution [33], and the popularity of a person in different networks is similar [34], that is, one person has similar social influence in different networks. But the degree of the node can only represent the number of its immediate neighbors, which could not represent the number of neighbors of its neighbors, that is, the number of second-order neighbors, if we want to express the structure of the node, we can define the node *k* degree. Therefore, the number of its *k*-order neighbors is also added to its degree, we recursively define the node's k-order degree d_A^k as (6):

$$\begin{cases} d_A^1 = d_A \\ d_A^k = d_A^{k-1} + \sum_{i=1}^{n} d_i \end{cases} \quad (6)$$

Therefore, we can compute the number of *n*-order neighbours of a node, which can show that if a node's influence and propagation ability can be an important feature of our node.

2) User's Clustering Coefficient

The clustering coefficient [35] represents the degree of node aggregation in a graph. The close relationship between the neighbors of the clustering coefficient, that is, the ratio of the neighbors to all possible relationships, as shown in (7):

$$C_i = \frac{2e_i}{k_i(k_i-1)} \quad (7)$$

Where e_i is the number of edges connect to node i, and k_i is the number of neighbors (degree of node i). For example, clustering coefficient reflects the closeness of node i's neighbors. In circle of classmates, circle of relatives, circle of colleagues or friends, the neighbors have high probability to become friends. In a cooperation network, people in the same circle are more likely to work together.

3) User's Eigenvector Centrality

Eigenvector centrality [36] reflects the number of node's neighbors and on the importance of each neighbor node. If x_i is the measure of the importance of node v_i, eigenvector centrality is defined as (8):

$$EC(i) = x_i = \lambda^{-1}\sum_{j=1}^{n} a_{ij}x_j \quad (8)$$

Where λ is a constant, $x=[x_1, x_2, x_3, \cdots, x_n]^T$, $x=\lambda^{-1}Ax$, $\mathbf{x}$ is the eigenvector of matrix $\mathbf{A}$ when eigenvalue is λ. The eigenvector centrality emphasizes the surrounding environment of the node (the number and quality of the neighbors of the node). Its essence is that the score of a node is the sum of the scores of its neighbors. The nodes can connect to many other important nodes enhance the importance of their own, high scores of nodes connect to a large number of general nodes, or connecting to a small number of other high-value nodes. Therefore, we can use eigenvector centrality as another feature of node in our solution.

E. Balancing Profile based Similarity and Network Structure based Similarity

Given that neither profile based similarity nor network structure based similarity can accurately represent the similarity between two nodes in different OSNs, we weight these two values and combine them so as to balance their importance. As (9) illustrates, the parameter α is a smooth factor that belongs to [0, 1]. As the value increases, much profile-based similarity can be taken into consideration, and vice versa. As the value drops, the situation will be opposite. MapMe gets the best result from tuning α, and the experiment result can be found in Section V.

$$sim(A,B)=\alpha\cdot sim_{\text{profile}}(A,B)+(1-\alpha)\cdot sim_{structure}(A,B) \quad (9)$$

Algorithm 1: User mapping across networks

Input: two OSNs G and G', and a set of known user pair M. θ is the threshold of similarity of two nodes.
Output: a set of inferred another link M'.

```
MappingUser(G, G', M)
{
    M'=∅;
    While (each candidate user pair (u_i^s, u_j^t) )
    {
      if (sim(u_i^s, u_j^t)> θ)
        M'= M' ∪ (u_i^s, u_j^t);
      u_i^s =BFSneighbor(u_i^s);
      u_i^t =BFSneighbor(u_j^t);
    }
  return M';
}
```

Consequently, our target has become a joint optimization problem. To solve that, we use the stochastic gradient descent (SGD) with the learning rate decay method for this optimization. The gradients are computed with back-propagation, and in our implementation, we first approximate the effect through instance sampling (node-node and node-content) in each training epoch. Then we add user pairs into the candidate set until the similarity is above a designated threshold. If two nodes' similarity is higher than threshold, then the two nodes are considered to be mapped. Finally, we adopt the similarity of threshold to determine whether the compared two anchor nodes belong to the same user or not. More details are shown in Algorithm 1.

V. EVALUATION

In this section, we give scheme evaluation which tests on both synthetic datasets and real-world datasets. The synthetic datasets are derived from Erdős-Rényi (ER) [37] random networks, Watts-Strogatz (WS) [38] small world networks and Barabási-Albert preferential attachment model (BA) networks [39]. Real-world datasets are captured from Aminer, Sina Weibo and ScienceNet's blog. Experiments are conducted on an Intel Core i7-2640M 2.80 GHz CPU, 32 GB RAM computer with a Matlab 2010b and Centos6.4 64-bit operating system. We employ Precision, Recall and F1 to evaluate the user account mapping methods. If a method finds an anchor link between OSNs, the method correctly recognizes a pair of mapping accounts; otherwise, the method makes a wrong mapping. Precision is the fraction of anchor links in the returned result that is correctly found. Recall is the fraction of the actual anchor links that are included in the returned result. F1 score is a weighted average of the precision and recall, F1=2* Precision* Recall/ (Precision+ Recall).

A. Synthetic Network Experiments

In this section, we use synthetic network to evaluate the performance of MapMe. Here, we set α=0 in (9) to test the performance with network structure-based similarity. Here, 10 pairs of networks are generated in the experiment. In ER and WS networks, *p* (the probability overlapping nodes with another network) equals to 0.1, 0.2, 0.3, 0.4 and 0.5, respectively. Similarly, in BA networks, *m* is the number of the added edges to the existing nodes, increased from 0 to 300 by 50.

TABLE I
RECALL RATE OF MAPME IN ER NETWORKS

Nodes	Known anchor links	P=0.1	P=0.2	P=0.3	P=0.4	P=0.5
10000	0.01	0.945	0.972	0.985	0.991	0.993
	0.02	0.951	0.975	0.992	0.994	0.997
	0.03	0.961	0.981	0.996	0.999	1
	0.04	0.978	0.982	0.997	1	1
	0.05	1	1	1	1	1
20000	0.01	0.953	0.979	0.986	0.992	0.995
	0.02	0.964	0.981	0.993	0.995	1
	0.03	0.973	0.986	0.997	1	1
	0.04	0.982	0.991	1	1	1
	0.05	0.994	1	1	1	1

Table I illustrates MapMe's performance in ER networks. Almost all the anchor links are identified with 4% of known anchor links. Table II shows MapMe is able to identify 99.7% anchor links with 2% known anchor links, and 99.9% anchor links are identified with 5% known anchor links in WS networks. When the probability overlapping nodes with another network *p* equals 0.1, the recall rate of MapMe in WS networks with 1% known anchor links is 69.1% and 73.1%, with 10000 and 20000 nodes, respectively. The recall rate of MapMe in ER networks with 1% known anchor links is 94.5% and 95.3%, with 10000 and 20000 nodes, respectively. The recall rate of MapMe in ER network is 23.8% higher than WS network on average with 1% known anchor links. Table III shows 99.9% anchor links are identified with 5% known anchor links in BA networks. In ER, WS and BA networks, almost all the anchor links are found with 5% known anchor links, which shows that MapMe can infer anchor links with a small fraction of known anchor links. Table IV shows the average precision, recall rate and F1 of MapMe in synthetic networks with different number of nodes. The ER network achieves the best precision and F1 values with 0.948 and 0.968 respectively. The lowest precision is 0.924 in these experiments, and the F1 value is 0.843. Therefore, MapMe is effective in synthetic networks.

TABLE II
RECALL RATE OF MAPME IN WS NETWORKS

Nodes	Known anchor links	P=0.1	P=0.2	P=0.3	P=0.4	P=0.5
10000	0.01	0.691	0.725	0.828	0.831	0.835
	0.02	0.783	0.931	0.982	0.989	0.996
	0.03	0.713	0.965	0.974	0.996	1
	0.04	0.970	0.995	0.998	1	1
	0.05	0.972	1	1	1	1
20000	0.01	0.731	0.763	0.989	0.995	0.998
	0.02	0.813	0.983	0.997	1	1
	0.03	0.994	1	1	1	1
	0.04	0.996	1	1	1	1
	0.05	0.999	1	1	1	1

TABLE III
RECALL RATE OF MAPME IN BA NETWORKS

Nodes	Known anchor links	m=50	m=100	m=150	m=200	m=250
10000	0.01	0.518	0.528	0.531	0.539	0.541
	0.02	0.621	0.964	0.971	0.979	0.981
	0.03	0.802	0.978	0.984	0.992	0.993
	0.04	0.874	0.986	0.989	0.993	0.995
	0.05	0.913	0.997	0.996	0.998	0.999
20000	0.01	0.845	0.921	0.914	0.943	0.948
	0.02	0.899	0.993	0.988	0.995	0.997
	0.03	0.911	0.996	0.999	1	1
	0.04	0.917	0.997	1	1	1
	0.05	0.918	0.999	1	1	1

TABLE IV
AVERAGE PRECISION, RECALL RATE AND F1 OF MAPME IN SYNTHETIC NETWORKS

Network (number of node)	Precision	Recall	F1
ER network (10000)	0.948	0.988	0.968
ER network (20000)	0.942	0.990	0.965
WS network (10000)	0.924	0.777	0.844
WS network (20000)	0.928	0.934	0.931
BA network (10000)	0.937	0.766	0.843
BA network (20000)	0.939	0.967	0.953

B. Social Media Network Experiments

In this section, we introduce the experiment datasets and then briefly introduce compared anchor link inferring methods. Finally, we give the experimental results and analysis.

1) Dataset Description

Our datasets are sourcing from the following three popular OSNs in China: Sina Weibo, Aminer and ScienceNet's blog. The timespan of datasets is from January to June 2015. The Sina Weibo dataset consists of 21 thousand user profiles and 769 thousand following relationships among them. The Aminer dataset consists of 17 thousand user profiles and 112 thousand following relationships among them. The ScienceNet's blog dataset consists of 16 thousand user profiles and 89 thousand following relationships among them. Since Aminer, Sina Weibo and ScienceNet's blog have lots of common users, our experiments are designed to map the common user accounts across the above three networks.

In these three networks, the anchor link between Sina Weibo and Aminer is 6,512, the anchor link between ScienceNet's blog and Sina Weibo networks is 4,096, and the anchor link between Aminer and ScienceNet's blog is 3,953.

2) Compared Methods

The social network datasets we are researching contains profile information and link information. We compare MapMe with the following methods for mapping user accounts across OSNs.

PNA [2]: This method extracts anchor adjacency features and latent topological features for mapping users across OSNs. The tensor decomposition techniques are used to predict anchor links on candidate mapped users. The generic stable matching method is used to prune the redundant anchor links in their solutions.

Method of Perito et al. [13]: Perito et al. find that a significant portion of user profiles could be linked by their nicknames. They estimate the uniqueness of a username to link profiles that have the same username and employ the language model and the Markov Chain to estimate username's uniqueness.

COSNET [16]: This method uses both local and global consistency among multiple OSNs. Features such as nickname, homepage, ego network and social status are used to connect user accounts across multiple OSNs. It utilizes an energy-based model to balance the importance of these features.

FRUI [27]: This is a friendship based method to map users across OSNs. FRUI employs mapped user accounts as seeds and utilizes seeds to identify other user accounts iteratively. This process does not need any control parameter.

MOBIUS [40]: This is a supervised learning method to use username features to identify corresponding users across OSNs. MOBIUS categorizes user behavioral patterns into human limitations, exogenous factors and endogenous factors, which are utilized to identify users across OSNs.

BIG-ALIGN [41]: BIG-ALIGN is an unsupervised bipartite graph alignment method, in which the user mapping problem is converted into an optimization problem.

UMA [42]: The Unsupervised multi-network alignment (UMA) method can partially align multiple networks by transitive network alignment and transitive network matching.

3) Experiment Results

This part shows the experimental results of all methods, including Precision, Recall and F1 with different sample rates.

Sample method analysis: in order to simulate network alignment, we use link sample rate η to control the number of anchor links, where η=0.1 means 10% users are mapped across the networks and η=1 means the users are totally mapped. In order to test whether different sample rates can influence the result of anchor link prediction, we give F1 score with different methods in Table V. Obviously, MapMe is outperforms other methods. The accuracy is improved with the increment of sample rates.

TABLE V

PERFORMANCE COMPARISON OF DIFFERENT METHODS FOR MAPPING USERS WITH DIFFERENT SAMPLING RATES

		Anchor Link Sampling Rate η									
	Methods	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
F1	PNA	0.005	0.026	0.052	0.083	0.286	0.397	0.418	0.427	0.444	0.469
	BIG-ALIGN	0.013	0.061	0.165	0.208	0.324	0.429	0.460	0.463	0.511	0.540
	UMA	0.018	0.070	0.274	0.273	0.397	0.523	0.537	0.501	0.556	0.569
	FRUI	0.213	0.270	0.379	0.425	0.502	0.585	0.713	0.791	0.909	0.913
	Method of Perito et al.	0.238	0.323	0.368	0.423	0.524	0.591	0.714	0.810	0.898	0.916
	MOBIUS	0.230	0.317	0.383	0.451	0.542	0.637	0.722	0.822	0.914	0.919
	COSNET	0.256	0.338	0.400	0.446	0.565	0.645	0.751	0.817	0.905	0.935
	MapMe	**0.280**	**0.375**	**0.442**	**0.496**	**0.614**	**0.716**	**0.775**	**0.842**	**0.940**	**0.964**

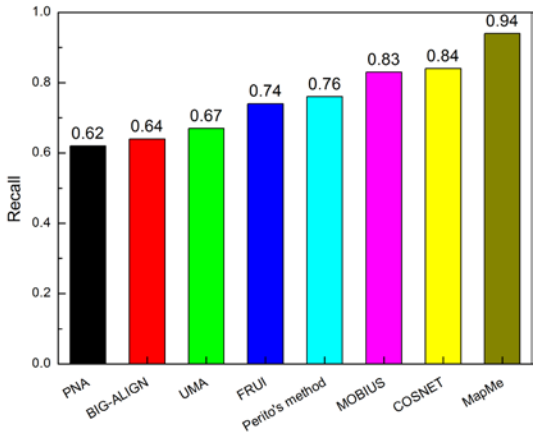


Fig. 2. Recall comparison with baseline methods

Comparison of Different Methods of Anchor Link Prediction: Fig.2 and Fig.3 shows recall and precision results under different link prediction methods, from which we can observe that MapMe achieves the best performance. MapMe's recall value is 32% higher than the value of PNA and 10% higher than the best solution COSNET in Fig. 2, while the precision value is 41% higher than PNA and 10% higher than the best proposal COSNET in Fig. 3.

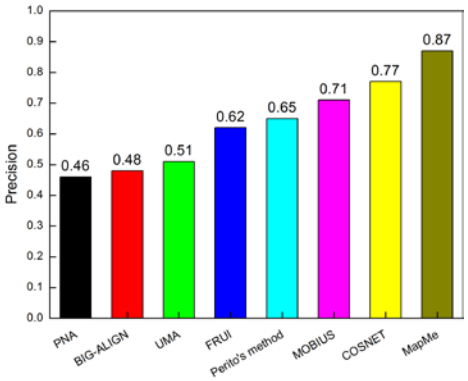


Fig. 3 Precision comparison with baseline methods

The MapMe Results at Different Sampling Rates: Fig. 4 shows Precision, Recall, F1 score of MapMe at different sample rates. We can clearly see that various indicators increase with the sample rate increase.

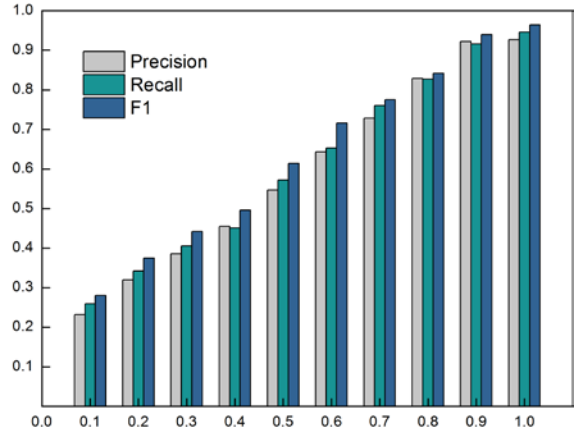


Fig. 4. Performance comparison based on sample rate η

Comparison of Various Methods in Precision, Recall, F1 Scores at Different Sample Rates: Fig. 5 shows the Precision, Recall, F1 scores of baseline methods with sampling rate 1, MapMe's precision, recall, F1 are 0.937, 0.946, 0.941 respectively, which is 10% higher on average than the state-of-the-art method.

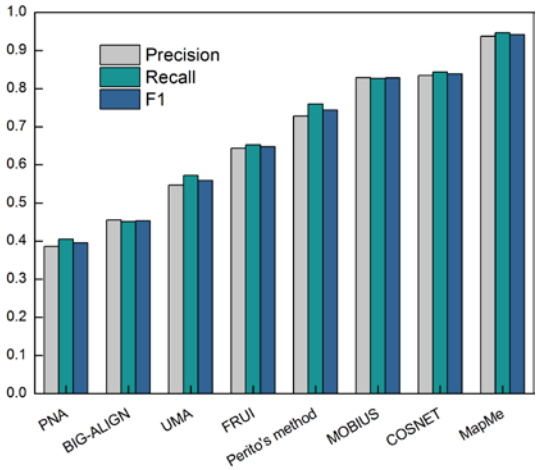


Fig. 5. Performance comparison with baseline methods

4) Parameter Sensitivity

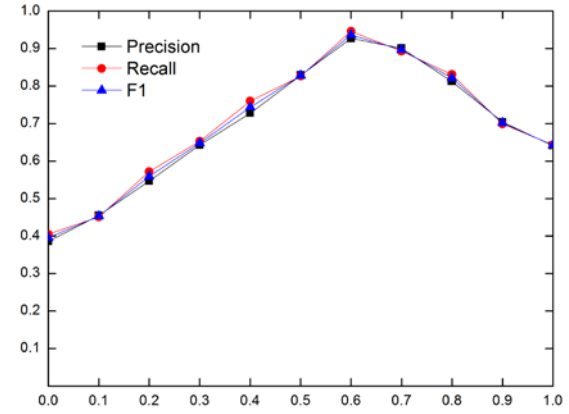


Fig. 6. Performance of MapMe with different α

In (9), α is a smooth factor for balancing the importance of the profile and structure similarity. Fig. 6 gives the Precision, Recall and F1 scores of MapMe with different α values. We observe that when α=0.6 the experiments get the best performance. It shows the profile based similarity is more important than structure-based similarity in these datasets. The reason is Aminer and ScienceNet's blogs are real name OSNs, and users intend to put real information in their profiles.

VI. Conclusion

In this paper, we proposed a novel joint model named MapMe, which employs both profile and topological features to map user accounts across OSNs. Within this model, MapMe extracts the profile features with Doc2vec methods and extracts node's *k* depth degree, clustering coefficient and eigenvector from social networks. MapMe derives pairs of similar nodes according to the similarity of profile and network structure features. In order to infer anchor links in sparse OSNs, MapMe partitions OSNs into small subgraphs with the spectral method. It matches similar subgraphs with graph distance and infers anchor links according to user similarity. Conducted experiments with synthetic and real datasets show that this solution is more accurate than the existing methods with 10% on average. In the future, we will apply our scheme into large-scale OSNs to prove its feasibility and advantages.

References

[1] "The 38th China Internet Development Report," [Online]. Available: http://www.cnnic.net.cn/gywm/xwzx/rdxw/2016/201608/W020160803204144417902.pdf, accessed Nov. 23, 2016.

[2] J. Zhang, W. Shao, S. Wang, X. Kong, and P. S. Yu, "PNA: Partial Network Alignment with Generic Stable Matching," in *IEEE International Conference on Information Reuse and Integration*, 2015, pp. 166–173.

[3] G. Qi, C. C. Aggarwal, and T. Huang, "Link prediction across networks by biased cross-network sampling," in *International Conference on Data Engineering*, 2013.

[4] P. M. Comar, P.-N. Tan, and A. K. Jain, "A framework for joint community detection across multiple related networks," *Neurocomputing*, vol. 76, no. 1, pp. 93–104, 2012.

[5] J. Leskovec, L. A. Adamic, and B. A. Huberman, "The Dynamics of Viral Marketing," *ACM Trans. Web*, vol. 1, no. 1, 2007.

[6] M. Al Hasan and M. J. Zaki, "A Survey of Link Prediction in Social Networks," in *Social Network Data Analytics*, C. C. Aggarwal, Ed. Boston, MA: Springer US, 2011, pp. 243–275.

[7] F. Benevenuto, T. Rodrigues, M. Cha, and V. Almeida, "Characterizing User Behavior in Online Social Networks," in *Proceedings of the 9th ACM SIGCOMM Conference on Internet Measurement*, 2009, pp. 49–62.

[8] Z.-K. Zhang, C. Liu, Y.-C. Zhang, and T. Zhou, "Solving the cold-start problem in recommender systems with social tags," *EPL (Europhysics Lett.*, vol. 92, no. 2, p. 28002, 2010.

[9] A. Narayanan and V. Shmatikov, "Myths and Fallacies of 'Personally Identifiable Information,'" *Commun. ACM*, vol. 53, no. 6, pp. 24–26, 2010.

[10] A. Gonzalez-Pardo, J. J. Jung, and D. Camacho, "ACO-based clustering for Ego Network analysis," *Futur. Gener. Comput. Syst.*, vol. 66, pp. 160–170, 2017.

[11] S. Tan, Z. Guan, D. Cai, X. Qin, J. Bu, and C. Chen, "Mapping Users across Networks by Manifold Alignment on Hypergraph," in *28th AAAI Conference on Artificial Intelligence*, 2014, pp. 159–165.

[12] R. Zafarani and H. Liu, "Connecting Corresponding Identities across Communities," in *International Conference on Weblogs and Social Media, Icwsm 2009, San Jose, California, Usa, May*, 2009.

[13] D. Perito, C. Castelluccia, M. A. Kaafar, and P. Manils, "How Unique and Traceable Are Usernames?," in *Privacy Enhancing Technologies: 11th International Symposium, PETS 2011, Waterloo, ON, Canada, July 27-29, 2011. Proceedings*, S. Fischer-Hübner and N. Hopper, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 1–17.

[14] 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 *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining*, 2013, pp. 495–504.

[15] T. Iofciu, P. Fankhauser, F. Abel, and K. Bischoff, "Identifying Users Across Social Tagging Systems," in *International Conference on Weblogs and Social Media, Barcelona, Catalonia, Spain, July*, 2010.

[16] Y. Zhang, J. Tang, Z. Yang, J. Pei, and P. S. Yu, "COSNET: Connecting Heterogeneous Social Networks with Local and Global Consistency," in *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2015, pp. 1485–1494.

[17] W. Cohen, P. Ravikumar, and S. Fienberg, "A comparison of string metrics for matching names and records," in *Kdd workshop on data cleaning and object consolidation*, 2003, vol. 3, pp. 73–78.

[18] A. Narayanan and V. Shmatikov, "De-anonymizing Social Networks," in *Security and Privacy Symposium on IEEE*, 2009, pp. 173–187.

[19] A. Malhotra, L. Totti, W. Meira Jr, P. Kumaraguru, and V. Almeida, "Studying user footprints in different online social networks," in *Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2012)*, 2012, pp. 1065–1070.

[20] R. Zafarani and H. Liu, "Connecting Users Across Social Media Sites: A Behavioral-modeling Approach," in *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2013, pp. 41–49.

[21] S. Liu, S. Wang, F. Zhu, J. Zhang, and R. Krishnan, "HYDRA: Large-scale Social Identity Linkage via Heterogeneous Behavior Modeling," in *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data*, 2014, pp. 51–62.

[22] O. Goga, H. Lei, S. H. K. Parthasarathi, G. Friedland, R. Sommer, and R. Teixeira, "Exploiting innocuous activity for correlating users across sites," in *Proceedings of the 22nd international conference on World Wide Web*, 2013, pp. 447–458.

[23] Y. Nie, Y. Jia, S. Li, X. Zhu, A. Li, and B. Zhou, "Identifying users across social networks based on dynamic core interests," *Neurocomputing*, vol. 210, pp. 107–115, 2016.

[24] Y. Cui, J. Pei, G. Tang, W.-S. Luk, D. Jiang, and M. Hua, "Finding email correspondents in online social networks," *World Wide Web*, vol. 16, no. 2, pp. 195–218, 2013.

[25] T. Man, H. Shen, S. Liu, X. Jin, and X. Cheng, "Predict Anchor Links across Social Networks via an Embedding Approach," 2016.

[26] S. Feng, D. Shen, Y. Kou, T. Nie, and G. Yu, "Anchor Link Prediction Using Topological Information in Social Networks," in *Web-Age Information Management: 17th International Conference, WAIM 2016, Nanchang, China, June 3-5, 2016, Proceedings, Part I*, B. Cui, N. Zhang, J. Xu, X. Lian, and D. Liu, Eds. Cham: Springer International Publishing, 2016, pp. 338–352.

[27] X. Zhou, X. Liang, H. Zhang, and Y. Ma, "Cross-Platform Identification of Anonymous Identical Users in Multiple Social Media Networks," *IEEE Trans. Knowl. Data Eng.*, vol. 24, no. 9, pp. 411–424, 2016.

[28] J. Tang, "AMiner: Mining Deep Knowledge from Big Scholar Data," in *Proceedings of the 25th International Conference Companion on World Wide Web*, 2016, p. 373.

[29] X. Kong, J. Zhang, and P. S. Yu, "Inferring Anchor Links Across Multiple Heterogeneous Social Networks," in *Proceedings of the 22Nd ACM International Conference on Information & Knowledge Management*, 2013, pp. 179–188.

[30] Q. V Le and T. Mikolov, "Distributed Representations of Sentences and Documents.," in *ICML*, 2014, vol. 14, pp. 1188–1196.

[31] R. C. Wilson and P. Zhu, "A study of graph spectra for comparing graphs and trees," *Pattern Recognit.*, vol. 41, no. 9, pp. 2833–2841, 2008.

[32] T. Opsahl, F. Agneessens, and J. Skvoretz, "Node centrality in weighted networks: Generalizing degree and shortest paths," *Soc. Networks*, vol. 32, no. 3, pp. 245–251, 2010.

[33] L. A. Adamic, R. M. Lukose, A. R. Puniyani, and B. A. Huberman, "Search in power-law networks," *Phys. Rev. E*, vol. 64, no. 4, p. 46135, 2001.

[34] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," *Proc. Natl. Acad. Sci.* , vol. 99, no. 12, pp. 7821–7826, Jun. 2002.

[35] P. Zhang, J. Wang, X. Li, M. Li, Z. Di, and Y. Fan, "Clustering coefficient and community structure of bipartite networks," *Phys. A Stat. Mech. its Appl.*, vol. 387, no. 27, pp. 6869–6875, 2008.

[36] P. Bonacich, "Some unique properties of eigenvector centrality," *Soc. Networks*, vol. 29, no. 4, pp. 555–564, 2007.

[37] A. Erdős, P. and Rényi, "On Random Graphs I," *Publ. Math.*, vol. 6, pp. 290–297, 1959.

[38] D. J. Watts and S. H. Strogatz, "Collective dynamics of |[lsquo]|small-world|[rsquo]| networks," *Nature*, vol. 393, no. 6684, p. 440, 1998.

[39] A.-L. Barabási and R. Albert, "Emergence of Scaling in Random Networks," *Science (80-.).*, vol. 286, no. 5439, p. 509 LP-512, Oct. 1999.

[40] R. Zafarani, L. Tang, and H. Liu, "User Identification Across Social Media," *ACM Trans. Knowl. Discov. Data*, vol. 10, no. 2, p. 16:1--16:30, 2015.

[41] D. Koutra, H. Tong, and D. Lubensky, "Big-align: Fast bipartite graph alignment," in *2013 IEEE 13th International Conference on Data Mining*, 2013, pp. 389–398.

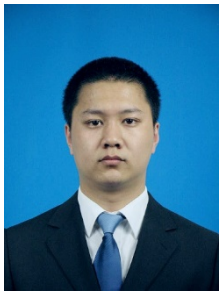
[42] J. Zhang and S. Y. Philip, "Multiple anonymized social networks alignment," in *Data Mining (ICDM), 2015 IEEE International Conference on*, 2015, pp. 599–608.

Jiang Jiangtao Ma is currently a Ph.D. candidate at the State Key Laboratory of Mathematical Engineering and Advanced Computing, China. He has been doing research work at Zhengzhou University of Light Industry since 2007. He received the B.S. degree and M.S. degrees in computer science from Zhengzhou University of Light Industry, China in 2004 and 2007, respectively. His research interests include machine learning, data mining and social network analysis. kitesmile2000@gmail.com

Yaqiong Qiao received the B.S. degree and M.S. degrees in Control Science and Engineering from Northwestern Polytechnical University, China in 2004 and 2007, respectively. She is currently a Ph.D. candidate at the State Key Laboratory of Mathematical Engineering and Advanced Computing, China. Her research interests include machine learning, data mining and social network analysis. qiaoyingqiong@126.com

Guangwu Hu is an associate professor of Shenzhen Insititue of Information Tech-nology. He received the Ph.D. degree in Computer Science and Tech-nology from Tsinghua University in 2014, then he became a postdoctor in the Graduate School at Shenzhen,Tsinghua University. His research interests include software defined networking, next-genera-tion Internet and Internet security. hugw@sziit.edu.cn

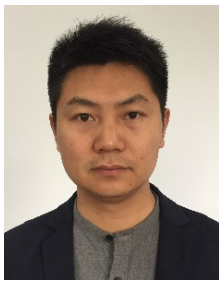
Yongzhong Huang received his Ph.D. degree in Computer Science from Zhengzhou University, China in 2007. He is currently a Professor at the State Key Laboratory of Mathematical Engineering and Advanced Computing, China. His research interests include high perfor-mance computing, big data analysis. 18600200718@163.com



Meng Wang received the B.S. degree in Computer Science from Tsinghua University, China in 2011, and the M.S. degree in Computer Science from China National Digital Switching System Engineering & Technological R&D Center, China in 2014. He is currently a Ph.D. candidate at the State Key Laboratory of Mathematical Engineering and Advanced Computing, China. His research interests include natural language processing, social network analysis and big data analysis. mengwangchn@outlook.com

Arun Kumar Sangaiah had received his Doctor of Philosophy (PhD) degree in Computer Science and Engineering from the VIT University, Vellore, India. He is presently working as an Associate Professor in School of Computer Science and Engineering, VIT University, India. His area of interest includes software engineering, computational intelligence, wireless networks, bio-informatics, and embedded systems. arunkumarsangaiah@gmail.com

Chaoqin Zhang is a Ph.D. candidate at the State Key Laboratory of Mathematical Engineering and Advanced Computing, China. Also, He is an associate professor of Zhengzhou University of Light Industry. His research interests include Internet architecture, data mining and network security. zhangcq@zzuli.edu.cn



Yanjun Wang received the B.S. degree and M.S. degrees in computer science from Zhengzhou University of Light Industry, China in 2004 and 2010, respectively. He is currently a Ph.D. candidate at the State Key Laboratory of Mathematical Engineering and Advanced Computing, China. He has been doing research work at Zhengzhou University of Light Industry since 2004. His research interests include big data, machine learning and internet user behavior analysis. 64502147@qq.com