

Social Friend Recommendation Based on Multiple Network Correlation

Shangrong Huang, Jian Zhang, *Senior Member, IEEE*, Lei Wang, and Xian-Sheng Hua, *Senior Member, IEEE*

Abstract—Friend recommendation is an important recommender application in social media. Major social websites such as Twitter and Facebook are all capable of recommending friends to individuals. However, most of these websites use simple friend recommendation algorithms such as similarity, popularity, or “friend’s friends are friends,” which are intuitive but consider few of the characteristics of the social network. In this paper we investigate the structure of social networks and develop an algorithm for network correlation-based social friend recommendation (NC-based SFR). To accomplish this goal, we correlate different “social role” networks, find their relationships and make friend recommendations. NC-based SFR is characterized by two key components: 1) related networks are aligned by selecting important features from each network, and 2) the network structure should be maximally preserved before and after network alignment. After important feature selection has been made, we recommend friends based on these features. We conduct experiments on the Flickr network, which contains more than ten thousand nodes and over 30 thousand tags covering half a million photos, to show that the proposed algorithm recommends friends more precisely than reference methods.

Index Terms—Social network alignment, friend recommendation, feature selection.

I. INTRODUCTION

SOCIAL networks have experienced explosive growth in the last decade. Social websites such as Twitter, YouTube and Flickr have billions of users who share opinions, photos and videos every day. Users make on-line friends through these social networks. One challenging issue is how to help these users to efficiently find new social friends. Social friend recommendation has therefore become a new research topic and several methods have been proposed to conduct recommendation efficiently [1], [2].

Content similarity (such as image visual similarity) has been a primary method of friend recommendation [1]. However, we argue that many other social aspects need to be explored to systematically build high-performance social friend recommendation, other than basing recommendation purely on content simi-

larity matching. Making friends is often based on the following social aspects: 1) Social environment, including where one lives and works [3]; 2) Social behaviours and actions, including one’s working performance, shopping habits, hobbies, and, importantly, interactions with one another [4], [5]. 3) Social status, such as gender, age, position, etc. [6] We summarize all these aspects as an individual’s “social role”. Here the term “social role” is the part that a person plays as a member of a particular society [7]. As stated in [7]: “In on-line social networks, people behave differently in social situations because they carry different latent social roles”. For example, a father and a child will respond differently when seeing a toy in a showcase at a shop. We believe that utilizing the individual’s social role information is a new research component for recommendation tasks.

These different social roles can be perceived in various social networks, such as a basketball-fan network, football-fan network, etc. These networks have the same set of nodes (each node represents one individual) but with different edge connections between nodes, because the meaning of the edges is different. Although each network represents one kind of relationship, its topology is not independent of other networks—in this paper, we define network topology as the arrangement of the edges of a network. This is because an individual’s various social roles are related to each other—a person’s hobbies are usually related to gender and age, while his/her friend circle is related to hobbies/positions, and so on.

We also observe that people have different social roles on the Web. For example, an individual who uses the substantial image sharing website Flickr¹ plays a number of on-line social roles such as that of a photo provider who shares photos, as well as tags that express personal feelings about the photos, a photo connoisseur, or simply one who wants to find friends with similar interests. These on-line social roles form different networks and these networks are related to one another. In this paper, we mine the correlations of these networks and propose a new approach for social friend recommendation. According to an individual’s social role, we recommend friends through alignment between different networks.

To leverage the correlations between networks, we present a social network in which the nodes of the graph are users and the edges stand for the relationships between users. Different kinds of relationships lead to networks with different topologies. Taking Flickr as an example, users may upload tags to describe their uploaded photos; if two users have become friends, then they are in each other’s “contact” list. In line with this information, we build a Flickr contact network in which the nodes are

Manuscript received January 19, 2015; revised June 18, 2015 and October 31, 2015; accepted December 02, 2015. Date of publication December 17, 2015; date of current version January 15, 2016. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Zhen Wen.

S. Huang and J. Zhang are with the Advanced Analytics Institute, School of Software, University of Technology Sydney, Sydney NSW 2007, Australia.

L. Wang is with the School of Computing and Information Technology, University of Wollongong, Wollongong NSW 2522, Australia.

X. Hua is with the Alibaba Group, Hangzhou 311121, China.

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TMM.2015.2510333

¹[Online]. Available: <http://www.flickr.com/>

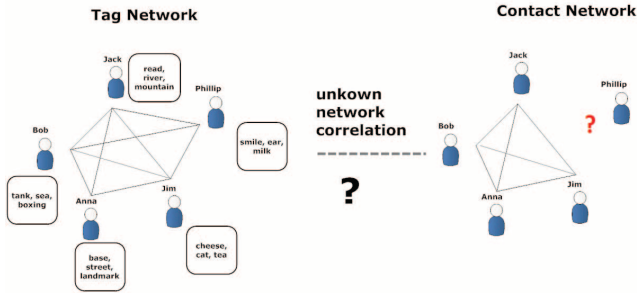


Fig. 1. Problem illustration: how to correlate the two networks and recommend friends to Phillip.

individuals and the edges represent whether they are friends or have “contact” with each other. Then we build a tag network in which the nodes are the same, but the edges represent the relationship of the tag set from each individual.

Fig. 1 illustrates the tag and contact network of a group of Flickr users. The left hand side of Fig. 1 is the Flickr tag similarity network of a small community of five people. The right hand side of Fig. 1 is the Flickr contact network. The tag and contact networks have the same nodes but different edges. We know the topologies of both networks except for the edges that connect with Phillip in the contact network. Phillip is new to the community and knows nobody else. He has already provided several tags that interest him via searching behaviours and is seeking new friends on Flickr. No correlations between the two networks have been built in Fig. 1, thus only simple content similarity recommendation based on the co-occurrence of tag can be applied for friend recommendation, whose accuracy is usually not satisfactory. Our problem is how to build the correlation of these two networks and make reliable friend recommendations.

“Correlation” between networks means that the topologies of different networks share similar properties. According to these similar properties, we can make inferences from one network to another. For example, if two nodes have a strong tie in the Flickr tag network, we might guess that they are also in each other’s contact list. However, we cannot say that they will be friends with each other in Flickr: Remember that the topologies of the tag and contact networks are not the same. To make more precise recommendation, we should determine how the two networks are correlated. In this paper, we propose to use network alignment methods. “Network alignment” here is defined as the action of mapping one network to another with a number of constraints/rules. It has been widely applied in the fields of bio-informatics [8] and computer vision [9]. In this paper, we take advantage of the study of network alignment in other fields, such as bio-informatics, to use as a new approach in social media.

To model the network correlations, we propose to align tag and contact networks through important tag feature selection. Here, an “important” feature is decided by whether it contributes to the correlation of the tag network with the contact network, or in other words, makes the topologies of the two networks more similar. The reason we select important features is that a person usually presents many social features in social networks, some

of which are attractive to others, and some of which are not very useful for building relationships. Determining those important features is convenient in enabling us to make faster decisions for friend making. It also saves both time and memory in the on-line recommendation system. To give a more specific example, a photographer uploads images to Flickr tags such as “natural animals”, “historical buildings”, “street views” and “people”. We view these tags as different feature words. The photographer may find that most of his friends in the Flickr network contact him because of the photos tagged with “natural animals” and “historical buildings”, rather than “street views” and “people”. This indicates that the first two feature words are more important than the last two for friend recommendation.

If two users in the tag network have a strong similarity in the selected features after the alignment, we can infer that they have a higher possibility of having a relationship in the contact network.

To make more precise friend recommendation, we also consider network structure preservation in our algorithm in addition to network alignment. Here “preservation” means that we do not significantly change the tag network structure before and after alignment. By preserving the tag network structure on Flickr, we reduce the over-fitting risk of our algorithm. A number of previous works have applied this concept for classification/clustering [10]. In this paper, we analyse its correctness mathematically and extend the idea to social media.

Roughly speaking, the algorithm is as follows. We align the tag and contact network by projecting the two networks into the lower dimensional spaces, in order to correlate them. We first project the contact network to its eigen-subspace, because the eigenspace usually carries important information from the original space. Then we project the tag network to another lower-dimensional subspace. The two subspaces of the tag and contact networks should, to some extent, match each other for more precise friend recommendation, compared to pure content similarity matching. One key point of our approach concerns important feature selection for the network matching. Details are given in Sections III and IV.

In summary, this paper makes the following contributions.

1. We propose a new friend recommendation method, based on network correlation, by considering the effect of different social roles.
2. To model the correlation between different networks, we develop a method that aligns these networks through important feature selection.
3. We also consider preserving the network structure for a more precise recommendation.
4. We conduct comprehensive experiments to show that the proposed method significantly improves the accuracy of friend-recommendation. To reduce the problem of biased data, we choose a very large dataset that is randomly crawled from Flickr.

The rest of the paper is organized as follows. Section II outlines related work. Section III introduces our framework and system model. Section IV gives the details of our algorithm. Section V details the performance of our method and analysis is made according to the result. Lastly, Section VI concludes our work.

II. RELATED WORK

In this section we introduce several research fields that directly relate to our work.

A. Friend Recommendation in Multimedia Environment

Recommendation systems have been widely used for a number of purposes such as item [11] or trip recommendation [12], media recommendation [13] and friend recommendation [14]–[16]. Different modalities of social data are used for synthesized recommendation, and a recent summary is given in [17]. For friend recommendation, several methods and applications have been published: [14] considers user-generated content on Twitter and develops a fast algorithm for real-time user-to-user similarity for follower/followee recommendation. It has been developed only for text similarity. [15] concentrates on mobile applications and makes friend recommendations by considering the influence of different social aspects such as locations and common interesting words. It defines a transition probability for friendship recommendation based on location neighbourhood and interesting word co-occurrence.

The work of [16] also combines social information from different layers. It takes both context and content information and associates it with domain knowledge. It takes user's implicit feedback to combine different kinds of information. A recent research direction called scalable recommendation [18] also incorporates different networks by introducing latent factors and maximizing the posterior distribution. In this paper, we mine the network structure and select important features via network alignment to obtain more precise friend recommendation.

B. Social Network Correlation

In this paper we study the correlations between different networks. Network or graph matching/correlation problems have been widely studied in some fields such as image retrieval [19], and bio-informatics [20]. In the field of social media, researchers have recently also proposed works that study the relationship of different networks. [21] is a pilot paper that studies the correlations between heterogeneous multimedia data. It studies how the data from different modalities are combined and used for cross-media retrieval, but it does not approach the problem from the aspect of networks. [22] studies the relationship of ratings in different networks by minimization assuming the latent factors of different domains, and minimizing a square loss function between the predicted rating and the real rating. [23] combines the score matrix from different domains using the matrix factorization method and applying a weighting matrix to mask the unobserved entries. Both [22] and [23] assume that latent factors exist to determine the observable ratings. In our work, we propose another approach that does not explicitly assume the inclusion of latent factors.

The work of [24] studies the matching of people's names and their social network identities, such as their Twitter account, with the help of common friends and the co-occurrence of words. It illustrates that pure text similarity matching has poor matching performance. By synthetically considering the text, the popularity and the relationships between people, the rate of making a correct match increases. This paper gives support to our idea that different social roles should be synthetically

considered for better recommendation. We develop this idea by considering the structure of networks and applying it to friend recommendation.

A more recent and related work for social media is given in [25]. In [25], three networks are first formed: user friendship network, tag network and image content network. Different relations are then defined within each network and between networks. According to these relations, the transition probability is defined and a random walk-based algorithm is developed to calculate the relative score between nodes. This propagation algorithm can be used for multi-purpose recommendation such as item/query/friendship, because the links between networks can be inferred. Compared with [25], our algorithm focuses on the use of a new network alignment method. Though both use multiple networks for link prediction, our proposed algorithm correlates networks with the same nodes and provides a mechanism for choosing important features. It provides a new viewpoint for interpreting the properties of the social network, compared with the pure propagation algorithm in [25].

C. Network Alignment

To find the correlations between different social networks, we propose to use network alignment methods. Several previous researches have considered the combination of different social networks for user behaviour prediction. [26] considers behaviours such as music listening and book reading for composite behaviour prediction. It uses a graphical model to build the relations of different networks. [27] utilizes the application installation information from mobile devices. It would be more satisfactory if these researches had utilized the ample topological information of networks to obtain a better result. In this paper, we consider the different social roles of individuals and use the topological information of different networks by alignment.

The concept of network alignment is applied in fields that study the relationships between big networks. In addition to the fields of bio-informatics and the image processing problem mentioned in the Introduction, it has also been applied to fields that deal with problems in large networks such as Internet network management [28]. In [28], alignment is used to find the co-occurrence of elements in different networks and reduce Internet data traffic. In this work, we extend the network alignment concept to social networks by considering the social roles of individuals for better recommendation.

D. Feature Selection

In this paper, we align different social networks through important feature selection. The initial motivation for feature selection is that the social data often contain many different features that are difficult to deal with, and most of the features are redundant except for specific tasks [29]. To deal with this problem, we usually apply feature extraction [30] or feature selection [31] methods. Feature selection is often preferred over extraction, because the selected features have more understandable physical meanings.

In [32], the authors provide a clustering method based on spectral embedding. Spectral embedding projects the data into a subspace, and chooses features that minimize the distance in each cluster on the projected subspace. However, it does not

consider the pairwise structure of the original data. [10] provides a model that considers both local and global structure preservation during projection. [10] induces the global preservation from linear kernel functions that can be applied in both the supervised and unsupervised case. The above works applied feature selection to image and bio datasets for clustering. Feature selection can also be applied to social media analysis; for example it can be used to choose important features from different domains to obtain cross-domain knowledge [33]. In this paper, we apply the concept of structure preservation by careful analysis to enable better use of social features for recommendation.

A number of previous works have combined the concept of feature selection and similarity network (kernels) alignment for other purposes. [34] studied the use of profile alignment and support vector machine for cellular localization. [35] applied network alignment to overcome the problem in which the locations of features vary from measurement to measurement for image matching. Our work differs from these previous works in two aspects: First, these works mostly concentrate on image processing and bio-informatics. In our proposed approach, we extend the concept to deal with larger and more complex social networks for social recommendation purposes. Second, most of the previous works are based on kernels that only apply the similarity information between users. Unlike these kernel-based methods, we utilize more detailed information that concerns not only the relationships between individuals, but also their social roles. This information is relatively easy to obtain in social media so we expect to achieve better results than pure kernel methods.

III. SYSTEM MODEL AND FRAMEWORK

In this section we present our framework. Details of the algorithm are given in Section IV.

A. Problem Statement and Notations

In NC-based SFR, there are two networks including a contact network, \mathcal{C} and a tag network \mathcal{T} (taking a real world example, \mathcal{C} stands for the contact network and \mathcal{T} for the tag similarity matrix on Flickr). \mathcal{C} and \mathcal{T} have exactly the same nodes but different topologies. As mentioned in the Introduction, the social roles of individuals are related to one another. \mathcal{T} shows an individual's interests and \mathcal{C} shows the friendship, so it is reasonable to assume that the topologies of tag and contact networks are correlated. In this paper, we propose a method to make more precise friend recommendations based on the correlations of different networks through their alignments.

When a new node comes into network \mathcal{T} , we know its links with other nodes in \mathcal{T} , but we do not know its links in network \mathcal{C} . Our research seeks to predict its links in \mathcal{C} . A real world example for this scenario is that when a new user comes into a social network, he/she may provide interesting keywords. The system should make friend recommendations for the new user, but traditional content similarity recommendation methods do not take the different aspects of social roles into account. In our approach, the alignment between social role networks is considered and thus a more comprehensive friend recommendation is obtained. We expect better performance using our algorithm.

Following are some of the notations used in this paper. In total, there are N nodes in \mathcal{C} and \mathcal{T} . The similarity matrix of

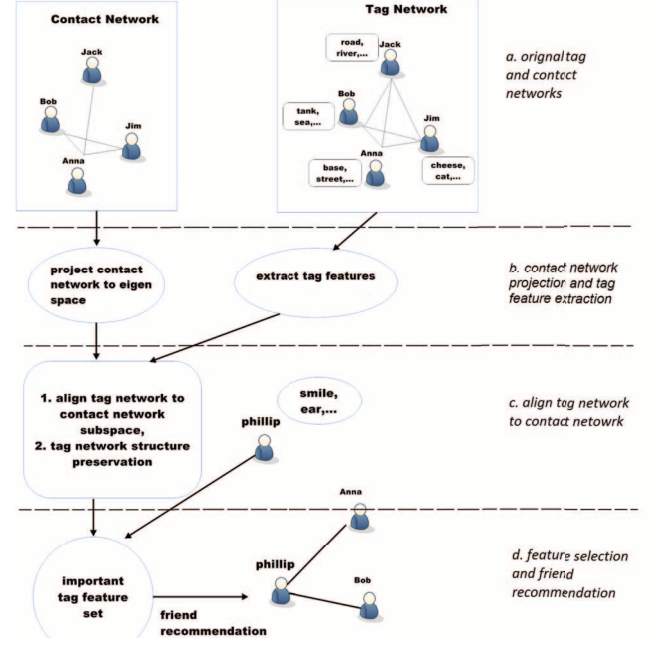


Fig. 2. Framework for our algorithm: (a) Input original tag and contact network. (b) Project the contact network to its eigen-space, extract features from tag network. (c) Align tag network with contact network in the subspace through feature selection. (d) New friends will be recommended based on selected important features.

network \mathcal{C} is given by $\mathbf{K} \in \mathbb{R}^{N \times N}$. In the above Flickr example, \mathbf{K}_{ij} is a binary number where “1” means $user_i$ and $user_j$ are friends, while “0” means they are not. $\mathbf{X} \in \mathbb{R}^{N \times F}$ is the feature matrix of network \mathcal{T} , where F stands for the dimensions of features to represent each node. In Flickr, F stands for the length of the whole dictionary of tags and \mathbf{X}_{ij} stands for whether $user_i$ uses tag_j . We also introduce the $N \times N$ matrix \mathbf{L} according to the tag similarity of each pair of users.

B. Our Framework

To make a prediction of network links according to the analysis in the Introduction, we will apply feature selection techniques to find the alignment of different networks that have the same nodes and different topologies.

In Fig. 2, we show the framework of our whole system. When we have the original tag and contact network as input [Fig. 2(a)], we first project the contact network to its eigenspace and extract tag features [Fig. 2(b)]. In our case, features are the tag words provided by photo uploaders. Then we align the tag network \mathcal{T} to the eigen-representation of the contact network \mathcal{C} [Fig. 2(c)] by considering network correlation and structure preservation. In the last step, we select a number of important word features from the whole feature set (which is composed of all the tag words). These important tag features illustrate the correlations between the tag and contact networks. In other words, these features make the tag network more similar to the contact network. When a new user with tags comes into the network, we can map him/her to the existing contact network based on how his/her tag features match the pool of those important features that have been selected previously. We can then see which users are closer to the new user, and these closer users are more likely to be his/her friends. Details of each step are explained in Section IV.

IV. NC-BASED SNC

In this section, we give the details of our algorithm by considering the correlation of two networks.

A. Approach Overview

We start by finding the alignment of two networks. In this paper we align different networks by selecting important features that capture the similarity of different networks. Taking the tag and contact networks as an example, we usually judge a person who might be our potential friend by only a few words. For example, when we find a friend on Facebook, we rarely read all of that person's posts because it is too time-consuming. Instead we only read the titles of several of his/her posts, to obtain a rough but to some extent accurate understanding of what he/she likes. Motivated by this phenomenon, we assume that an individual's friend-making decisions are determined by a small number of features from a large feature set. The whole feature set may contain tags, photos, comments, geo information, etc., and according to the previous discussion, we assume that based on a relatively small number of features, different networks should be well-aligned. In this paper, we choose tag features for correlation with contact information, and the idea can be extended to different kinds of networks.

We choose features from two aspects: In the first aspect, features are chosen that correlate two networks well. We fix one network \mathcal{C} and align the topologies of the other networks such as \mathcal{T} in a subspace. Details for the network correlation are given in Section IV-C.

In the second aspect, in addition to network alignment, we choose features that preserve the original structure of the modified network \mathcal{C} . In other words, nodes that are close to each other in the original network should also be close in the modified network, thus the network alignment does not change the pairwise similarity between nodes. The effectiveness of pairwise similarity preservation has been shown in [10], [36]. By doing this we may predict links for new nodes for network \mathcal{C} according to the existing links in modified network \mathcal{T} more precisely. We will discuss this more carefully in Section IV-D.

Section IV-B shows some small but non-trivial methods for filtering noise and redundancies. Section IV-C and IV-D illustrate the details of the proposed algorithm, Section IV-E gives the solution as well as the complexity analysis of the proposed algorithm.

B. Bag of Words and Feature Extraction

In the alignment of the tag and contact networks, we treat tag words as features. The tag data crawled from social websites such as Flickr usually contain much noise and thus data refinement is required for a better recommendation result. After removing explicit stop features such as “a” and “the”, as well as features that too often or too seldom appear in Flickr tags, we build the vocabulary of the tags.

To calculate the tag feature matrix \mathbf{X} , we adopt the widely used TF-IDF method [37]. Apart from counting the number of words that each user has used as the tags of his/her photos, TF-IDF assumes that features that seldom appear carry more information. Under this assumption, TF-IDF diminishes the

weight of features that occur frequently in the dataset and increases the weight of features that occur rarely. By calculating the TF-IDF of each word, we build the feature matrix \mathbf{X} .

C. Network Alignment

As mentioned in Section IV-A, we first consider minimizing the gap between graphs \mathcal{C} and \mathcal{T} by selecting important features. Assume we have a feature selection matrix \mathbf{W} , then the selected features from the whole feature matrix can be expressed as \mathbf{XW} . To make the user-user similarity between \mathcal{C} and \mathcal{T} as small as possible, we have the following formulation:

$$\begin{aligned} \min_{\mathbf{W}} \quad & \| \mathbf{XW}\mathbf{W}^T\mathbf{X}^T - \mathbf{K} \|_F^2 \\ \text{subject to : } \quad & \mathbf{W} \in \{0, 1\}^{F \times r} \\ & \mathbf{W}\mathbf{1}_{F \times 1} = \mathbf{1}_{F \times 1} \end{aligned} \quad (1)$$

where \mathbf{K} is the similarity matrix of \mathcal{C} defined in III-A. $\mathbf{W} \in \{0, 1\}^{F \times r}$ means that \mathbf{W} can only be chosen from $\{0, 1\}$ and $\mathbf{W}\mathbf{1}_{F \times 1} = \mathbf{1}_{F \times 1}$ means that the sum of each row in \mathbf{W} is exactly 1. These two constraints ensure that each feature can only be chosen once so that we do not have a linear combination of features. This is a discrete problem and is hard to solve. For computational efficiency following [38] we drop the integer constraint of \mathbf{W} .

Also, the user similarity in (1) measured by inner product makes the dimension too high for optimization. In [38], the problem is approximated by the following:

$$\min_{\mathbf{W}} \| \mathbf{XW} - \mathbf{V} \|_F^2 \quad (2)$$

where $\mathbf{V} \in R^{N \times r}$ stands for the r eigenvectors corresponding to the largest eigenvalues of \mathbf{K} . r also stands for the dimension of the subspace in which the data is projected. The work in [38] shows that solving the optimization problem in (2) is a good approximation to solving the problem in (1). By adding a regularization term we have the following optimization problem:

$$\min_{\mathbf{W}} \| \mathbf{XW} - \mathbf{V} \|_F^2 + \lambda \| \mathbf{W} \|_{2,p} . \quad (3)$$

The regularization term $\| \mathbf{W} \|_{2,p}$ forces \mathbf{W} to be sparse. Traditional methods such as [10], [38] often use $l_{2,1}$ norm for sparse representation. Here we use $l_{2,p}$ norm following the ideas in [36]. The value of p controls the number of zeros in \mathbf{W} , and thus determines whether the features selected by \mathbf{W} are more discriminative. It also affects the final recommendation results.

In this way, we make the gap between two networks as small as possible.

D. Structure Preservation

Considering only the gap minimization between two networks might lead to a problem whereby the structure of the modified network is greatly changed, leading to an over-fitting problem. One good way to overcome this problem is to preserve the neighbourhood relationship of the network during projection. That is, the distance between any two nodes before and after the projection should not change much. [39] provides a good deduction to preserve the structure of the dataset, and [10] has shown through experiments that by preserving the

pairwise similarity of the original data structure, the clustering and classification performance is improved, compared to pure gap minimization. [10] studies the problem in image and biological data. In this subsection we extend the idea of pairwise similarity relationship to the field of social network. First we calculate the structure of the original dataset.

1) *Data Structure Representation*: The structure of a network can be expressed as the pairwise similarity between two different nodes in the graph. For the tag feature matrix, it can be expressed as the semantic meaning of words between users.

After we obtain feature matrix \mathbf{X} , a simple way to calculate the similarity between two users is to count their co-occurrence features. However, this method is too simple and sometimes fails to catch the similarity between users. For example, a user with the tag “river” might have more topics in common with a user with “mountain” than another user with “basketball”. The above-mentioned simple method cannot capture this similarity. In this paper, we use WordNet² to calculate the semantic similarity between different features [40]. WordNet groups words into sets of synonyms, and different synonyms are connected with hypernyms/hyponyms (as a simple example, *apple* is a hyponym of *plant* and *food*, *plant* and *food* are hypernyms of *apple*). According to [41], different methods of obtaining feature similarity measurements can be applied. Two different features will be given a relation score between 0 and 1.

Other tag feature similarity methods have recently been seen in the field of natural language processing, such as Microsoft Deep Semantic Similarity Model (DSSM)³ and Google similarity distance [42]. As WordNet has been widely applied in many previous works, we apply it in our work for similarity calculation by following the literature.

After obtaining the relation score of different features, we calculate the similarity between users to obtain L . For $user_i$ having a tag set of $word_{1i}, word_{2i}, \dots, word_{si}$, $user_j$ having a tag set of $word_{1j}, word_{2j}, \dots, word_{tj}$, we have the relation score matrix $\mathbf{E}_{ij} \in s \times t$. and the similarity between i and j is given by

$$S_{ij} = \sum_s \sum_t \mathbf{E}_{ijst}. \quad (4)$$

2) *Structure Preservation*: After we represent the data structure and the similarity matrix \mathbf{S} , we study how to preserve the network structure during the alignment.

To minimize the pairwise-distance changes, we define the data after projection as $\mathbf{A} = \mathbf{W}^T \mathbf{X}^T$. Also we set a pairwise distance matrix of all samples as \mathbf{L} , which can be deduced from \mathbf{S} of (4).

We then want to minimize the change of in-dimension distance of all the r dimensions

$$\begin{aligned} \min_{\mathbf{W}} \sum_{j=1}^n \sum_{i=1}^n \mathbf{A}_{1i} \mathbf{A}_{1j} \mathbf{L}_{ij} + \sum_{j=1}^n \sum_{i=1}^n \mathbf{A}_{2i} \mathbf{A}_{2j} \mathbf{L}_{ij} \\ + \dots + \sum_{j=1}^n \sum_{i=1}^n \mathbf{A}_{ri} \mathbf{A}_{rj} \mathbf{L}_{ij}. \end{aligned} \quad (5)$$

The first term of the above equation can be written in a matrix form as follows:

$$\begin{aligned} \sum_{j=1}^n \sum_{i=1}^n \mathbf{A}_{1i} \mathbf{A}_{1j} \mathbf{L}_{ij} \\ = \sum_{i=1}^n \mathbf{A}_{1i} \mathbf{L}_{i1} \mathbf{A}_{11} + \mathbf{A}_{1i} \mathbf{L}_{i2} \mathbf{A}_{12} + \dots + \mathbf{A}_{1i} \mathbf{L}_{in} \mathbf{A}_{1n} \\ = \left[\sum_{i=1}^n \mathbf{A}_{1i} \mathbf{L}_{i1}, \sum_{i=1}^n \mathbf{A}_{1i} \mathbf{L}_{i2}, \dots, \sum_{i=1}^n \mathbf{A}_{1i} \mathbf{L}_{in} \right] (\mathbf{A}_{row1})^T \\ = (\mathbf{A}\mathbf{L})_{row1} (\mathbf{A}_{row1})^T = (\mathbf{A}\mathbf{L}\mathbf{A}^T)_{11}. \end{aligned} \quad (6)$$

Now we consider the inner distance of other dimensions. Similarly, we have

$$\begin{aligned} \sum_{j=1}^n \sum_{i=1}^n \mathbf{A}_{2i} \mathbf{A}_{2j} \mathbf{L}_{ij} &= (\mathbf{A}\mathbf{L}\mathbf{A}^T)_{22} \\ &\dots \\ \sum_{j=1}^n \sum_{i=1}^n \mathbf{A}_{ri} \mathbf{A}_{rj} \mathbf{L}_{ij} &= (\mathbf{A}\mathbf{L}\mathbf{A}^T)_{rr}. \end{aligned} \quad (7)$$

By adding together the r terms listed above, we show that formulation 5 can be expressed as the trace of a matrix

$$\begin{aligned} \min_{\mathbf{W}} \sum_{j=1}^n \sum_{i=1}^n \mathbf{A}_{1i} \mathbf{A}_{1j} \mathbf{L}_{ij} + \dots + \sum_{j=1}^n \sum_{i=1}^n \mathbf{A}_{ri} \mathbf{A}_{rj} \mathbf{L}_{ij} \\ = \min_{\mathbf{W}} \text{tr}(\mathbf{A}\mathbf{L}\mathbf{A}^T) = \min_{\mathbf{W}} \text{tr}(\mathbf{W}^T \mathbf{X}^T \mathbf{L} \mathbf{X} \mathbf{W}). \end{aligned} \quad (8)$$

E. Solutions and Complexity Analysis

1) *Solutions*: By combining the optimization problems in Section IV-C and IV-D, we have the following optimization problems:

$$\min_{\mathbf{W}} \|\mathbf{X}\mathbf{W} - \mathbf{V}\|_F^2 + \mu \text{tr}(\mathbf{W}^T \mathbf{X}^T \mathbf{L} \mathbf{X} \mathbf{W}) + \lambda \|\mathbf{W}\|_{2,p} \quad (9)$$

where λ and μ are regularization parameters. We limit the value of p to be above 1 to make the problem convex. The $l_{2,p}$ -norm constraint makes the problem hard to solve. According to [10], it can be approximated by $\mathbf{W}^T \mathbf{B} \mathbf{W}$, where \mathbf{B} is a diagonal matrix and defined as follows:

$$\mathbf{B}_{ii} = \frac{p}{2 \|\mathbf{W}_{row-i}\|_2^{2-p}}. \quad (10)$$

So (9) is formulated as follows:

$$\min_{\mathbf{W}} \|\mathbf{X}\mathbf{W} - \mathbf{V}\|_F^2 + \mu \text{tr}(\mathbf{W}^T \mathbf{X}^T \mathbf{L} \mathbf{X} \mathbf{W}) + \lambda \text{tr}(\mathbf{W}^T \mathbf{B} \mathbf{W}). \quad (11)$$

It can be solved efficiently by iteratively solving \mathbf{W} and \mathbf{B} . When \mathbf{B} is fixed, \mathbf{W} can be obtained by setting the derivation of (11) to zero, and then we obtain \mathbf{W} as follows:

$$\mathbf{W} = (\mathbf{X}^T (\mathbf{I} + \mu \mathbf{L}) \mathbf{W} + \lambda \mathbf{B})^{-1} \mathbf{X}^T \mathbf{V}. \quad (12)$$

²[Online]. Available: <http://wordnet.princeton.edu>

³[Online]. Available: <http://research.microsoft.com/en-us/projects/dssm/>

Equation (12) requires the calculation of matrix inverse of size $F \times F$. If F is much large than N , the time complexity is too large. Note that (12) has the equivalent form as follows:

$$\mathbf{W} = \frac{1}{\lambda} \mathbf{B}^T \left(\frac{1}{\lambda} (\mathbf{I} + \mu \mathbf{L}) \mathbf{X} \mathbf{B} \mathbf{X}^T + \mathbf{I} \right)^{-1}. \quad (13)$$

Equation (13) calculates the inverse of a $N \times N$ matrix. When the sample size N is relatively small, we can apply this equation.

After obtaining \mathbf{W} , we update \mathbf{B} by (10). When p is no less than 1, it is a convex problem. By iteratively updating \mathbf{W} and \mathbf{B} , (9) theoretically converges to a global optimum, according to [10]. For most cases in experiments, it converges to a reasonably small range within five rounds of iterations.

2) *Feature Selection*: The next step is to choose important features according to the optimal \mathbf{W} . From (1) we see that each row of \mathbf{W} corresponds to one feature. The larger the norm of this row, the more important is the role this feature plays in aligning network \mathcal{T} to network \mathcal{C} . The norms of rows that are nearly zero mean that the corresponding features make almost no contribution to the alignment, so we rank the norm of rows of \mathbf{W} according to their norms in descending order, and choose features according to this order. The top features are considered to be the most important for friend prediction. In the experimental section, we verify that by considering weight we achieve a slightly better performance.

3) *Friend Recommendation*: How do we make recommendations according to the important features for a new user with some tag words coming into the network? As mentioned previously, we select these features based on the alignment of the tag network to the contact network, and these important features illustrate the correlations of the contact and tag networks. We therefore calculate the similarities between the important tag features of the new user and those of the existing users, because the important features pool contains the tags that make great contributions to the correlation between tag and contact networks. Therefore, this similarity of important features implies the relationship of the new user to existing users in the contact network. The more similar the important tag set, the closer the two users should be. After ranking the tag similarity of the new user and the members already in the networks, we choose the most similar K members as possible friends to recommend to the new user.

The whole process is shown in Algorithm 1.

Algorithm 1 Proposed NC based SFR

Input:

tag feature matrix \mathbf{X} , contact matrix \mathbf{K} , tag feature vector of the new user \mathbf{x} , number of friends K

Output:

Friend recommendation list

- 1: Determine λ , μ and p via cross validation on training set
- 2: Calculate the tag relationship matrix \mathbf{L}
- 3: Calculate \mathbf{V} by eigen-decomposition of Laplacian of \mathbf{K}
- 4: Initialize \mathbf{B} with identity matrix \mathbf{I}
- 5: **repeat**

6: Calculate \mathbf{W} by (12) or (13)

7: Calculate \mathbf{B} by (10)

8: **until** *Convergence*

9: Calculate the norm of each row of \mathbf{W} . Rank the norms in a descending order.

10: Choose important features from top of the ranking list.

11: Calculate the similarities between the important features of the new user and those of the existing users.

12: Top K similar users are recommended as friends to the new user.

4) *Parameter Choice*: Step 1 in Algorithm 1 determines the best value for parameters λ , μ and p . In practice, the best value of parameters μ , λ and p is determined by employing the cross validation method on the training set. We choose the best value of the parameters so that we obtain the highest friend recommendation accuracy.

5) *Complexity and Large Scale Data Suitability*: One pre-processing step before Algorithm 1 is the calculation of computing tag similarity matrix \mathbf{L} . Assuming the time for each similarity function of WordNet takes τ seconds, because \mathbf{L} is symmetric, the time to calculate relation score matrix \mathbf{E} is $F \times (F - 1) \times \tau / 2$.

To calculate \mathbf{L} from \mathbf{E} by (4), it takes $N \times N \times s \times t$ sum operation.

One time-consuming job in algorithm 1 is the $N \times r$ eigenvector matrix calculation in step (3), of which the complexity is $\mathcal{O}(rN^2)$.

Another time-consuming job in Algorithm 1 is the inverse of the matrix in (12) or (13). The complexity at each iteration is $\mathcal{O}(\min\{N, F\}^3)$.

From the above discussion, the complexity of the algorithm is mainly determined by $\max((\min\{N, F\}^3), rN^2)$. Because r is usually small for the data we deal with, the complexity does not increase with the increase in the scale of the data when the number of features is fixed. Our algorithm therefore fits large scale social networks that contain millions of users. This is a good property of our algorithm.

F. Expansion to Multiple Networks

From Sections IV-A to IV-E we have studied the problem of two-network correlation. However, our algorithm is not limited to two networks but can be applied to many different correlated networks. As we mentioned in Section I, each person plays a different social role in different social networks. Until now, we have studied the simplest case of two-network alignment, but in real social life people play many different social roles and multiple networks can be formed. We can choose features from these different networks to reflect the impact of various social roles for friend recommendation. Next we provide a simple solution for multi-network alignment.

The system model is shown in Fig. 3. We take the Flickr network as an example: Apart from the tag features in Flickr, we also have image features, comment features, geo features, and favourite photo features. These different features may form different networks. The question is, how do we predict a new user's friendship when different network topologies are given? In other

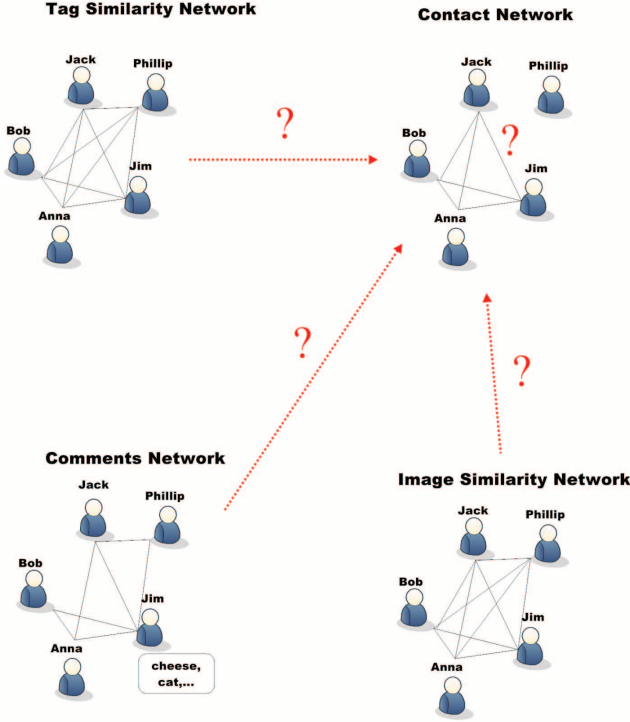


Fig. 3. How to correlate multiple networks.

words, how do we correlate more networks to make a combined inference?

Assume we have M networks with the same nodes but different topologies. These topologies are not the same but they are related to each other. Each other network should have a projection matrix \mathbf{W}_m , and each network has its own relationship matrix \mathbf{L}_m , $1 \leq m \leq M - 1$. Thus, the optimization problem can be formulated as follows:

$$\min_{\mathbf{W}_m} \sum_{m=1}^{M-1} (\|\mathbf{X}\mathbf{W}_m - \mathbf{V}\|_F^2 + \mu_m \text{tr}(\mathbf{W}_m^T \mathbf{X}^T \mathbf{L}_m \mathbf{X} \mathbf{W}_m) + \lambda_m \|\mathbf{W}_m\|_{2,p_m}) \quad (14)$$

where λ_m and μ_m are predefined regularization parameters. Equation (14) can be solved by iteratively fixing $m - 2$ \mathbf{W}_m s projection matrix and optimizing the last matrix. In this way we obtain the local optimum.

V. EXPERIMENTS

In this section we conduct extensive experiments to show the effectiveness of our proposed method, as well as to illustrate a number of interesting properties. We provide a case study on the Flickr dataset which we have crawled ourselves. First we give a brief introduction of our social media dataset, and then we discuss our algorithm from different aspects.

A. Dataset

1) *Building Dataset*: We crawled a social network from the big image sharing site Flickr. The dataset should not be biased nor too sparse. To alleviate the sparsity friendship problem of our dataset, we crawled the users in groups. To reduce bias, we crawled groups randomly. As the dataset is quite large, a

TABLE I
DATASET STATISTICS

Users	10000
Groups	2000
Photos	543,754 photos from 10000 users
Contact	145,684 friend links among users
Tags	35,574 words after filtering

relatively unbiased dataset can be obtained. A “group” in Flickr is a user-created album that relates to a topic, such as “Sydney”, “bike”, “autumn”, etc. Members of this group can upload photos to this group for sharing. Together we have crawled the data of 10000 users from 2000 groups. In our experiment, we randomly crawled five users in each group.

For each user, we crawled all their photos, and the tags of each photo. For the same users, we crawled their contact information to form the contact network. In Flickr, contact information is obtained according to whether a user has added another user to his/her friend list, or vice versa. We crawled all the contacts between any of the two users in our dataset. A short summary of our dataset is given in Table I.

In our experiments, the features in Table I are tag words. Other features such as image features and geo features might be integrated in future works.

2) *Training and Testing*: For each round of experiments, a 5-fold cross-validation test is conducted. We divide our dataset equally and randomly into five sets. Each time we choose four sets as training data to determine the important features and leave one set to be used as testing data for friend recommendation based on the previously chosen features. In each round we run five times of training and testing. In total, we conduct four rounds of 5-fold cross-validation, and the final result is obtained by averaging the outcomes of the four rounds.

B. Experimental Settings

Our task is to make precise contact information prediction. In this way, when a new user comes into the social network, we recommend new friends to him/her by providing key words in which he/she is interested. By considering different aspects of social roles, we correlate the tag feature network and contact network through alignment for friend recommendation.

1) *Parameter Settings*: In friend recommendation, assume for each user we recommend K friends to him/her. We use the existing contact information as the ground truth for training and testing. Parameters λ , μ and p are determined on the training set by a fourfold cross validation to find the best. The ranges for these parameters are: $\lambda \in 10^{[-2:1:3]}$, $\mu \in 10^{[-2:1:3]}$, and $p \in [1, 1.5, 2, 2.5, 3]$.

2) *Metrics*: We use the method stated in Section IV-E3 to recommend friends to new users. We may use the precision and recall metrics to show the effectiveness of the proposed algorithm. In our experiment, precision is defined as the correctly recommended friends divided by all the recommended users, and recall is defined as the correctly recommended friends divided by the number of all truly existing friends.

One problem for precision and recall is that when one becomes large, the other usually becomes small, so we use F-measure to combine the two

$$F = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}. \quad (15)$$

C. Reference Methods

We use several reference methods to show the improvements of our proposed algorithm in friend recommendation. They are: 1) random tag similarity, 2) SVM, 3) on-line collaborative filtering (OLCF) [43], and 4) Relational Domain Recommendation (RDR) [44].

The first is the simplest random tag similarity comparison. We randomly choose a number of features and recommend friends of each user on tag similarity.

The second method is an SVM-based method. It implements the “one-against-one” approach for multi-class classification [45]. We choose SVM as the reference method because in social media, friend recommendation can be viewed as a classification problem, where each user is classified as “recommended” or “not recommended”. In our experiment, we crawled the data of 10000 users from 2000 Flickr groups. Each group has five users and we assume each group to be one labelled class. Because each group is very small, we assume the members in each group are friends with each other. Note that this assumption holds only in the SVM method for experimental convenience. For training we choose four users in each group and test whether the last user can be classified correctly. If it is correctly classified, we assume we have an accurate friend recommendation result.

The third method is OLCF. The collaborative filtering method is widely used in recommender systems. It fills the blank entries of the user-item matrix. In our experiment we use a model-based collaborative filtering method to determine one user's favour of each feature, and then calculate the cosine similarity of the whole feature space between users. By ranking these similarities we recommend friends.

A number of traditional model-based collaborative filtering methods face the problem that when a new user comes, the whole latent space has to be updated [11]. In this paper, we apply an online-updating collaborative filtering method as the reference method [43].

Last, we consider a multi-network based algorithm for comparison. When considering social multiple network problems, transition probability propagation is a method that is frequently used [44], [25]. We choose [44] as a reference method for the following reasons: 1) It considers the relationships of different networks, which is similar to our idea, though in [44], different networks have different nodes; 2) It uses the information of other networks for recommendation, which again has some similarities with ours. [44] enhances the links in one network and between different networks using a random walk propagation method. After a sufficient number of walks, it obtains the modified link weights between each user pair. We use the weights for friend recommendation. [25] also uses a random walk based method but considers more kinds of relationships. Due to space limitations, we do not detail it here but leave it for future work.

D. Experimental Results

1) *The Effect of Number of Chosen Words*: First we study the algorithm performance with different number of features. We change the selected number of features from 300 to 12000, and test first the precision for each number of features. We recommend the first 20 most similar users as friends for each user

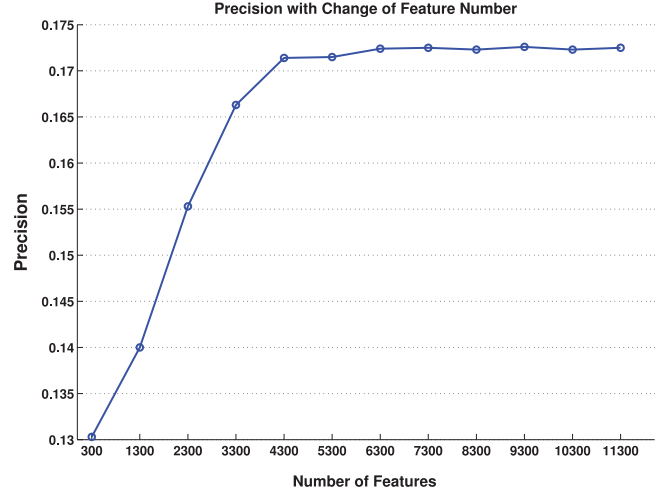


Fig. 4. Recommendation precision (recommend 20 people) increases sharply with the increase of features when the feature number is small; after a turning point, it becomes stable.

with certain number of features. We determine parameters in (9) with a five cross validation procedure for the best value. The result of precision is shown in Fig. 4.

From Fig. 4, we see that the performance of the algorithm increases quickly when the number of features is relatively small. There is a turning point when we choose the number of features around 3000. When the feature number exceeds 4500, the performance does not increase greatly. This means that when a user looks for friends with only specific tag words but not all the tags, we can make a relatively precise recommendation. This coincides with our intuition that when people makes friends, they do not consider all the aspects of others. Instead, they might only concentrate on a few important aspects of others to make a decision.

2) *Feature Selection vs. Selection Plus Re-Weighting*: After the calculation of Section IV-E2 we obtain the importance of each feature. We then rank the features according to their importance, and we obtain a list of features. At the top of the list are those features of most importance. Two methods can be applied in the next step: First, we choose important features from the top of the list; Second, we again choose features from the top of the list and consider their weightings for friend recommendation. Below are the results for pure feature selection and selection with re-weighting.

We fix the number of selected features as 4500 and the total number of the dataset as 10000, and change the number of recommended friends. The result is shown in Fig. 5.

We see that there is a slight improvement when considering the weights of each feature. This tells us that the weights of features play a role in recommendation. On the other hand, in some cases, we can only choose important features without re-weighting for simplicity, but the performance will not decrease much significantly.

From Fig. 5 we observe that the recommendation performance reduces with the increase in the number of recommended friends (from 5 to 25). This is mainly because the contact network is generally sparse, so the increase in numbers leads to an inevitable decrease in precision.

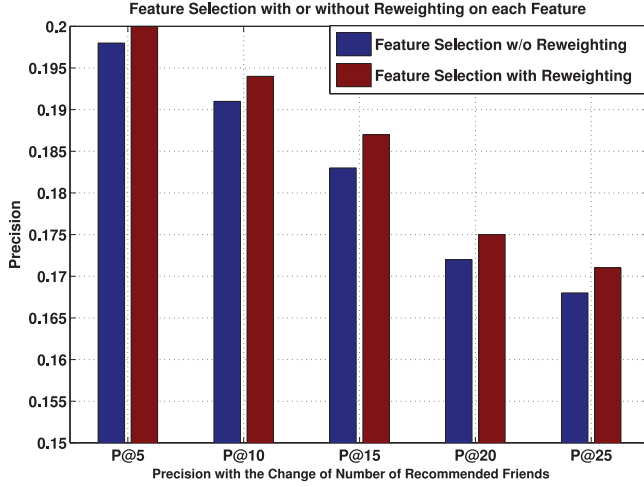


Fig. 5. Comparison of feature selection and re-weighting with different numbers of recommended friends. The feature re-weighting has slightly better performance.

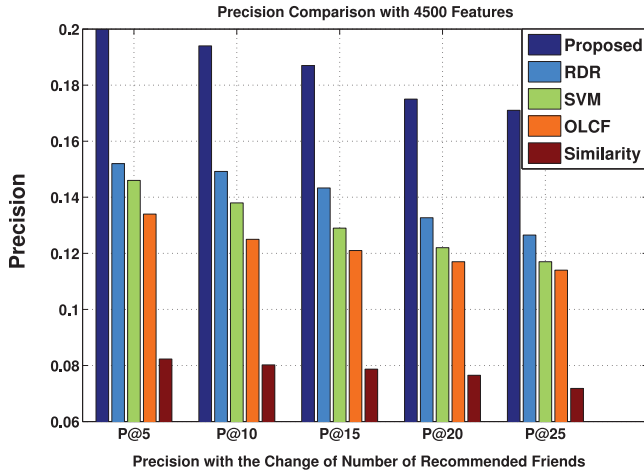


Fig. 6. Comparison with different reference methods at 4500 features. The proposed method has achieved the highest recommendation precision.

3) *Comparison of Different Methods for Precision:* In this experiment we compare the proposed method with all the reference methods mentioned in Section V-C.

We fix the number of features for the proposed method as 4500. Now we compare the Precision Measure performance of different methods with the reference methods mentioned in Section V-C. We change the number of recommended friends K from 5 to 25 for a more complete comparison. The resulting histogram is given in Fig. 6.

From Fig. 6 we see that no matter which K we choose, the proposed method outperforms all other reference methods. The random tag similarity method has the lowest recommendation precision.

The collaborative filtering method OLCF has slightly lower performance than SVM. The reason might be noise and redundancy are added by filling the user-feature matrix, thus affecting its performance.

The propagation-based algorithm RDR has the second best performance. It enriches the user-to-user link by random walk in contact and other networks and thus has a better performance

TABLE II
EXAMPLE OF SELECTED AND DISCARDED FEATURES

selected features	Manhattan, Eiffel, sunshine, skyline, Rome, Tank, ...
discarded features	Nikon, outdoor, Luis Valadares people, beautiful, water, ...

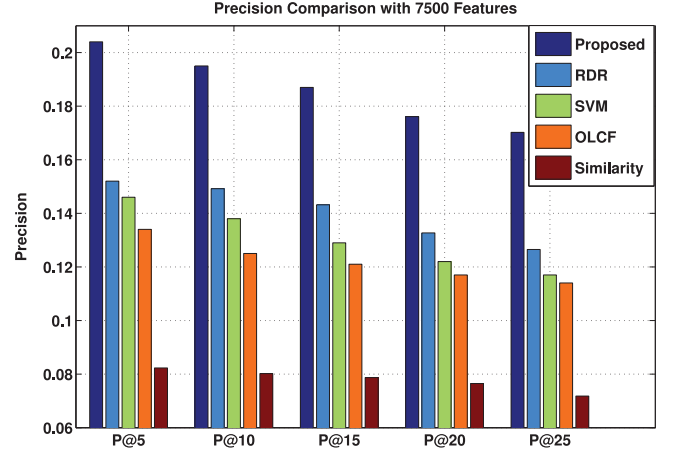


Fig. 7. Precision performance when applying 7500 features: Our method has the best performance and the result shows little change, compared with 4500 features.

than SVM. However, it lacks a mechanism to consider what is important in a friend-making decision, which has been carefully considered in our proposed algorithm.

The proposed method has the best friend prediction accuracy. This is because we correlate the tag information with the contact network. We choose the most important features when people make decisions about making friends with each other.

To illustrate this more clearly, we list several selected features and several discarded features in Table II

From Table II we see that some relatively “general” words such as “outdoor”, “beautiful” do not help much in friend recommendation. The helpful words are the more specific ones such as “Eiffel” and “Tank”. On the other hand, words that are too specific such as “Canon” or “LuisValadares” do not help much, either.

The result when we increase the number of selected features to 7500 is shown in Fig. 7.

Fig. 7 shows that the performance of the proposed method does not increase greatly with the increase in features, when the number of features is relatively large. The reason for this is that we have already included all the most important features in the previous 4500 features. The rest can be regarded as redundancies for friend recommendation.

4) *Comparison for Recall and F-Measure:* As mentioned before, recall and F-measure are also common metrics to measure the effectiveness of recommendation algorithms. Figs. 8 and 9 give the results of our proposed method and the reference methods. Here we fix the number of recommended users as 5, the number of features as 4500, and change the total number of users from 5000 to 10000.

From Figs. 8 and 9 we see that the system performance changes when the number of users is changed, and our proposed method always has the best performance.

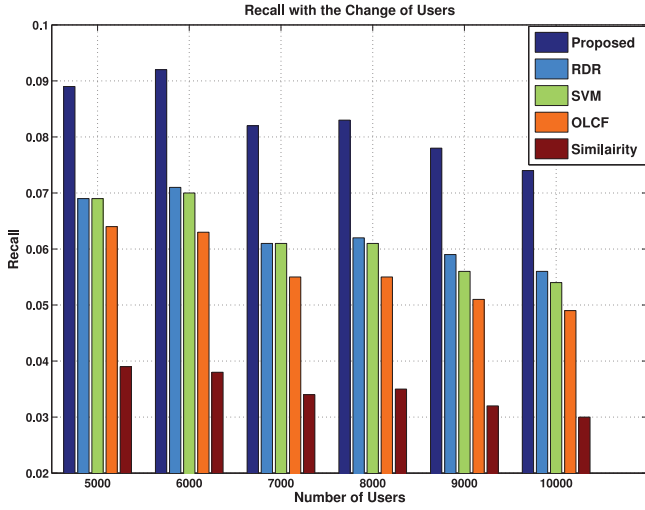


Fig. 8. Recall comparison: our algorithm has the highest recall, which means that it can discover the maximum number of real friends, compared with others.

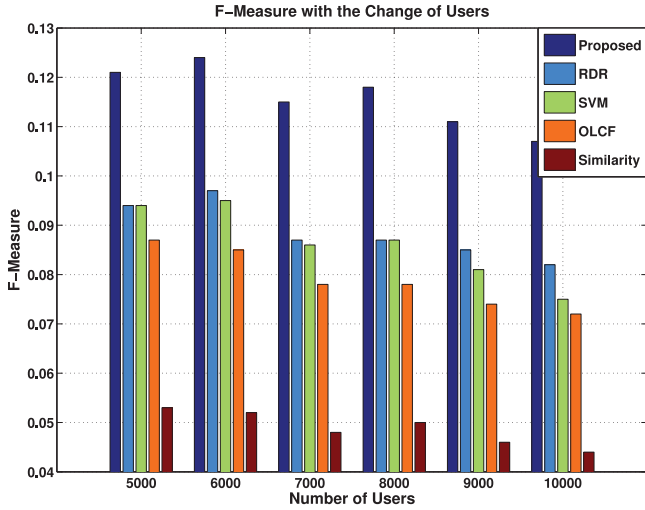


Fig. 9. F measure result: the alignment-based method has a higher F-measure than the reference methods.

5) *Effect of Parameters*: In this experiment we first study the effect of parameter μ . It adjusts the weight between the alignment of two networks and the preservation of the data structure. Again we fix the feature number as 4500 and K as 20, λ is fixed as 1. The result is given in Fig. 10(a).

Fig. 10(a) shows that with the increase of μ , the prediction accuracy first increases to a maximum point, then decreases. This phenomenon tells us that both the network alignment and network structure preservation play a significant role in important feature selection. Maximum friend recommendation precision is reached when we balance both of the two well.

We then consider the influence of the sparsity of \mathbf{W} . A different choice of p controls the sparsity of the projection matrix \mathbf{W} .

Fig. 10(b) shows that the sparsity of \mathbf{W} has an influence on friend recommendation accuracy. As p increases, the recommendation accuracy first stays almost unchanged, and then decreases quickly.

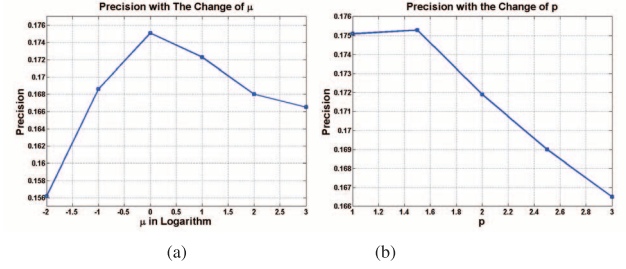


Fig. 10. μ decides the proportion of network alignment and structure preservation in our algorithm; p controls the sparsity of the alignment matrix \mathbf{W} . They both influence recommendation precision.

VI. CONCLUSION AND FURTHER WORKS

In this paper, we study the friend recommendation problem from the viewpoint of network correlation. A person has many different social roles on-line. In each social role, he/she makes different friends, and these different social roles form different social networks. To consider the effect of different social roles, we propose a network alignment method to find the correlations between networks. The second aspect we take into account is pairwise user similarity preservation to maintain the original data structure.

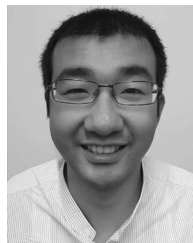
Experimental results by aligning tag and contact networks have shown that the proposed NC-based SFR outperforms other methods in friend recommendation and achieves the highest precision in friend prediction. We find that a small number of features can align the tag network to the contact network well, and can provide sufficient information for friend recommendation. Both network alignment and social network structure preservation play an important role in our task.

In future, we will further develop our algorithm in the following aspects: 1) In this paper, we consider different social networks to have similar structures and we handle them using similar methods. In experiments we align only two networks. We will extend the idea of network alignment to many networks, and consider the individual properties of these networks to make better recommendations. 2) We will apply the idea of network correlation for applications other than friend recommendation.

REFERENCES

- [1] J. Chen, W. Geyer, C. Dugan, M. Muller, and I. Guy, "Make new friends, but keep the old: Recommending people on social networking sites," in *Proc. ACM CHI*, Apr. 2009, pp. 201–210.
- [2] S. Wan, Y. Lan, J. Guo, C. Fan, and X. Cheng, "Informational friendship recommendation in social media," in *Proc. ACM SIGIR*, Jul. 2013, pp. 1045–1048.
- [3] E. Barnett and M. Casper, "A definition of 'social environment'," *Amer. J. Public Health*, vol. 91, p. 465, Mar. 2001.
- [4] R. J. Rummel, *The Conflict Helix: Principles and Practices of Interpersonal, Social, and International Conflict and Cooperation*. New Brunswick, N.J, USA: Transactions Books, 1991.
- [5] M. Weber, *The Nature of Social Action*. Cambridge, U.K.: Cambridge Univ. Press, 1991.
- [6] R. Brym, *Sociology: Your Compass for a New World, Brief Edition*. Boston, MA, USA: Cengage Learning, 2009.
- [7] Y. Zhao, G. Wang, P. S. Yu, S. Liu, and S. Zhang, "Inferring social roles and statuses in social networks," in *Proc. ACM KDD*, Aug. 2013, pp. 695–703.
- [8] G. Klau, "A new graph-based method for pairwise global network alignment," *BMC Bioinform.*, vol. 10, pp. S59:1–S59:9, Jan. 2009.

- [9] D. Conte, P. P. Foggia, C. Sansone, and M. Vento, "Thirty years of graph matching in pattern recognition," *Int. J. Pattern Recog. Artif. Intell.*, vol. 18, pp. 265–298, May 2004.
- [10] X. Liu, L. Wang, J. Zhang, J. Yin, and H. Liu, "Global and local structure preservation for feature selection," *IEEE Trans. Neural Netw. Learning Syst.*, vol. 25, no. 6, pp. 1083–1095, Nov. 2014.
- [11] H. Ma, H. Yang, M. R. Lyu, and I. King, "SOREC: Social recommendation using probabilistic matrix factorization," in *Proc. ACM CIKM*, Nov. 2008, pp. 931–940.
- [12] Y.-Y. Chen, A.-J. Cheng, and W. H. Hsu, "Travel recommendation by mining people attributes and travel group types from community-contributed photos," *IEEE Trans. Multimedia*, vol. 15, no. 6, pp. 1283–1295, Oct. 2013.
- [13] S. D. Roy, T. Mei, W. Zeng, and S. Li, "Socialtransfer: Cross-domain transfer learning from social streams for media applications," in *Proc. ACM MM*, Oct. 2012, pp. 649–658.
- [14] J. Hannon, M. Bennett, and B. Smyth, "Recommending twitter users to follow using content and collaborative filtering approaches," in *Proc. ACM RecSys*, Sep. 2010, pp. 199–206.
- [15] N. Li and G. Chen, "Multi-layered friendship modeling for location-based mobile social networks," in *Proc. IEEE MobiQuitous*, Jul. 2009, pp. 1–10.
- [16] X. Xie, "Potential friend recommendation in online social networking," in *Proc. IEEE/ACM CPSCom*, Dec. 2010, pp. 831–835.
- [17] I. Fernandez-Tobias, I. Candtador, M. Kaminskas, and F. Ricci, "A generic semantic-based framework for cross-domain recommendation," in *Proc. 2nd Int. Workshop Inform. Heterogeneity Fusion Recommender Syst.*, 2011, vol. 9, pp. 25–32.
- [18] M. Jiang, P. Cui, W. Zhu, and S. Yang, "Scalable recommendation with social contextual information," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 11, pp. 2789–2902, Nov. 2014.
- [19] M. Gori, M. Maggini, and L. Sarti, "Exact and approximate graph matching using random walks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 7, pp. 1100–1111, Jul. 2005.
- [20] M. H. Goodfellow, "Biochemical network matching and composition," in *Proc. EDBT/ICDT Workshops*, Mar. 2010, pp. 40:1–40:7.
- [21] Y.-T. Zhuang, Y. Yang, and F. Wu, "Mining semantic correlation of heterogeneous multimedia data for cross-media retrieval," *IEEE Trans. Multimedia*, vol. 10, no. 2, pp. 221–229, Feb. 2008.
- [22] W. Pan, E. W. Xiang, and Q. Yang, "Transfer learning in collaborative filtering with uncertain ratings," in *Proc. AAAI Conf. Artif. Intell.*, 2010, vol. 10, pp. 230–235.
- [23] B. Li, Q. Yang, and X. Xue, "Can movies and books collaborate? cross-domain collaborative filtering for sparsity reduction," in *Proc. IJCAI*, 2009, vol. 9, pp. 2052–2057.
- [24] G. won You, S. won Hwang, Z. Nie, and J.-R. Wen, "Socialsearch: Enhancing entity search with social network matching," in *Proc. EDBT*, Mar. 2011, pp. 515–519.
- [25] D. Liu, G. Ye, C.-T. Chen, S. Yan, and S.-F. Chang, "Hybrid social media network," in *Proc. ACM MM*, Oct. 2012, pp. 659–668.
- [26] E. Zhong, W. Fan, and Q. Yang, "User behavior learning and transfer in composite social networks," *ACM Trans. Knowl. Discovery Data*, vol. 8, pp. 6:1–6:32, Feb. 2014.
- [27] W. Pan, N. Aharoni, and A. S. Pentland, "Composite social network for predicting mobile apps installation," in *Proc. AAAI*, Aug. 2011, pp. 821–827.
- [28] C. Kreibich and J. Crowcroft, "Efficient sequence alignment of network traffic," in *Proc. ACM IMC*, Oct. 2006, pp. 307–312.
- [29] R. E. Bellman, *Dynamic Programming*. Mineola, NY, USA: Courier Dover, 2003.
- [30] I. T. Jolliffe, *Principal Component Analysis*. New York, NY, USA: Springer, 1986.
- [31] H. Liu and H. Motoda, *Computational Methods for Feature Selection*. London, U.K.: Chapman & Hall, 2008.
- [32] D. Cai, C. Xiang, and X. He, "Unsupervised feature selection for multi-cluster data," in *Proc. ACM SIGKDD*, Jul. 2010, pp. 333–342.
- [33] Y. Yang, Z. Ma, A. G. Hauptmann, and N. Sebe, "Feature selection for multimedia analysis by sharing information among multiple tasks," *IEEE Trans. Multimedia*, vol. 15, no. 3, pp. 661–669, Apr. 2013.
- [34] J. Guo, M. Man-Wai, and S.-Y. Kung, "Eukaryotic protein subcellular localization based on local pairwise profile alignment SVM," in *Proc. 16th IEEE Signal Process. Soc. Workshop Mach. Learn. Signal Process.*, Sep. 2006, pp. 391–396.
- [35] P. Kosir, R. DeWail, and R. A. Mitchell, "A multiple measurement approach for feature alignment," in *Proc. IEEE Nat. Aerospace Electron. Conv.*, May 1995, vol. 1, pp. 94–101.
- [36] Y. Yang, Z. Xu, G. Liu, Z. Ma, and N. Sebe, "Global structural feature selection with sparsity for multimedia data understanding," *ACM MM'13*, pp. 537–540, Oct. 2013.
- [37] H. C. Wu, R. W. P. Luk, L. K. F. Wong, and K. L. Kwok, "Interpreting TF-IDF term weights as making relevance decisions," *ACM Trans. Inform. Syst.*, vol. 26, pp. 13:1–13:27, Jun. 2008.
- [38] Z. Zhao, L. Wang, H. Liu, and J. Ye, "On similarity preserving feature selection," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 3, pp. 619–632, Mar. 2013.
- [39] X. He and P. Niyogi, "Locality preserving projections," *Neural Inform. Process. Syst.*, vol. 16, p. 153, 2004.
- [40] P. Nithiya, V. Vidhya, and L. Ganesan, "Development of semantic based information retrieval using word-net approach," in *Proc. IC-CCNT*, Apr. 2010, pp. 320–323.
- [41] Y. Jin, L. Khan, L. Wang, and M. Awad, "Image annotations by combining multiple evidence & Wordnet," in *Proc. ACM MM*, Nov. 2005, vol. 15, pp. 706–715.
- [42] R. Cilibrasi and P. M. B. Vitanyi, "The Google similarity distance," *IEEE Trans. Knowl. Data Eng.*, vol. 19, no. 3, pp. 370–383, Mar. 2007.
- [43] S. Rendle and L. S. Thieme, "Online-updating regularized kernel matrix factorization models for large-scale recommender systems," in *Proc. ACM RecSys*, Oct. 2008, pp. 251–258.
- [44] M. Jiang, P. Cui, F. Wang, Q. Yang, W. Zhu, and S. Yang, "Social recommendation across multiple relational domains," in *Proc. ACM CIKM*, Oct. 2012, pp. 1422–1431.
- [45] S. Knerr, L. Personnaz, and G. Dreyfus, "Single-layer learning revisited: A stepwise procedure for building and training a neural network," *Neurocomput.: Algorithms, Architectures Appl.*, vol. 68, pp. 41–50, 1990.



Shangrong Huang received the B.Sc. degree in telecommunication from Zhejiang University, Hangzhou, China, in 2005, the M. Sc. degree in information and communication technology from Darmstadt University of Technology, Darmstadt, Germany, in 2009, and is currently working toward the Ph.D. degree at the University of Technology Sydney, Sydney, Australia.

His research interests include recommendation system, social medias and big data.



Jian Zhang (S'95–M'98–SM'04) received the B.Sc. degree from East China Normal University, Shanghai, China, in 1982, the M.Sc. degree in computer science from Flinders University, Adelaide, SA, Australia, in 1994, and the Ph.D. degree in electrical engineering from the University of New South Wales (UNSW), Sydney, NSW, Australia, in 1999.

From 1997 to 2003, he was with the Visual Information Processing Laboratory, Motorola Labs, Sydney, Australia, as a Senior Research Engineer, and later became a Principal Research Engineer and a Foundation Manager with the Visual Communications Research Team. From 2004 to July 2011, he was a Principal Researcher and a Project Leader with National ICT Australia, Sydney, Australia, and a Conjoint Associate Professor with the School of Computer Science and Engineering, UNSW. He is currently an Associate Professor with the Advanced Analytics Institute & School of Software, Faculty of Engineering and Information Technology, University of Technology Sydney, Sydney, Australia. He has authored or coauthored more than 100 paper publications and book chapters, and has six issued patents filed in the U.S. and China. His current research interests include multimedia processing and communications, image and video processing, machine learning, pattern recognition, media and social media visual information retrieval and mining, human-computer interaction, and intelligent video surveillance systems.

Dr. Zhang is an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY and the EURASIP *Journal on Image and Video Processing*. He was the General Co-Chair of the International Conference on Multimedia and Expo in 2012 and the Technical Program Co-Chair of the IEEE Visual Communications and Image Processing 2014.



Lei Wang received the Ph.D. degree from Nanyang Technological University, Singapore, in 2004.

He is now with the School of Computing and Information Technology, University of Wollongong, Wollongong, NSW, Australia, as an Associate Professor. His research interests include machine learning, pattern recognition, and computer vision, particularly feature selection, model selection, kernel-based learning methods, image categorization, and content-based image retrieval.



Xian-Shen Hua (M'05–SM'14) received the B.S. degree in 1996, and the Ph.D. degree in applied mathematics in 2001, both from Peking University, Beijing, China.

He joined Microsoft Research Asia, Beijing, China, in 2001, as a Researcher. He was a Principal Research and Development Lead in Multimedia Search for the Microsoft search engine, Bing, Redmond, WA, USA, from 2011 to 2013. He was a Senior Researcher with Microsoft Research Redmond, Redmond, WA, USA, from 2013 to 2015. He

became a Researcher and Senior Director of the Alibaba Group, Hangzhou, China, in April of 2015, leading the Multimedia Technology Team in the Search Division. He has authored or coauthored more than 250 research papers and has filed more than 90 patents. His research interests include multimedia search, advertising, understanding, and mining, as well as pattern recognition and machine learning.

Dr. Hua served or is now serving as an Associate Editor for the IEEE TRANSACTIONS ON MULTIMEDIA and ACM *Transactions on Intelligent Systems and Technology*. He served as a Program Co-Chair for IEEE ICME 2013, ACM Multimedia 2012, and IEEE ICME 2012, as well as on the Technical Directions Board of the IEEE Signal Processing Society. He was one of the recipients of the 2008 MIT Technology Review TR35 Young Innovator Award. He was the recipient of the Best Paper and Best Demonstration Awards at ACM Multimedia 2007, the Best Poster Award at the IEEE International Workshop on Multimedia Signal Processing 2008, the Best Student Paper Award at the ACM Conference on Information and Knowledge Management 2009, the Best Paper Award at the International Conference on MultiMedia Modeling 2010, the Best Demonstration Award at ICME 2014, and Best Paper Award of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY in 2014.