Social Friend Recommendation Based on Multiple Network Correlation

Shangrong Huang, Jian Zhang, Lei Wang,and Xian-Sheng Hua

***Abstract*—Friend recommendation is an important recommender application in social Media. Major social websites such as Tweet 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 do not satisfy the majority of users. 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 recommendation. NC-based SFR is characterized by two key components: 1) We align related networks by selecting important features of each network. 2) Network structure should be maximally preserved before and after network alignment. After important feature selection 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 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[1][2].

Content similarity (such as image visual similarity) has been a primary clue for 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 similarity matching. People making friends 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, which entail various expectations that society puts on them."*. From our point of view, we believe that utilizing the individual's different social role information would be a new research component for recommendation tasks. In this paper, We define network topology as the arrangement of edges of a network.

These different social roles can be perceived in different 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 are different. Although each network represents one kind of relationship, its topology is not independent of other networks. 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 can also observe one's different social roles on web. For example, for an individual who uses the big image sharing website Flickr[1], he/she plays different on-line social roles such as a photo provider who shares his photos, as well as tags about his/her feelings about the photos, a photo connoisseur, or simply one who wants to find some friends who have some photos he/she also has interest. These individual's on-line social roles form different networks and these networks are related to each other. 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 correlations between different networks, we first present a social network as a graph in which the nodes of the graph are users and the edges stand for the relationships between users. Taking the contact and tag imformation on Flickr as an example, we build a contact graph in which the nodes are individuals and the edges represent their friendships. We then build a tag graph, in which the nodes are the same, but the edges represent the similarity of the tag set from each individual.

Figure 1 illustrates the tag and contact network of a group of Flickr users. The left hand side of Figure 1 is the Flickr tag similarity network of a small community with five people. The right hand side of Figure 1 is the Flickr contact network. We know the topologies of both networks except the edges connected with Phillip in the contact network. Phillip is new

S. Huang and Jian Zhang are with the Advanced Analytics Institute, School of Software, Univerity of Technology Sydney, Australia.

L. Wang is with School of Computer Science and Software Engineering, University of Wollongong, Australia.

X. Hua is with Microsoft Research, Redmond, USA

[1] http://www.flickr.com/

to the community and knows nobody else. He has already provided several tags that interest him via searching behavious and is seeking new friends on Flickr. No correlations between the two networks have been built in Figure 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 recommendation.

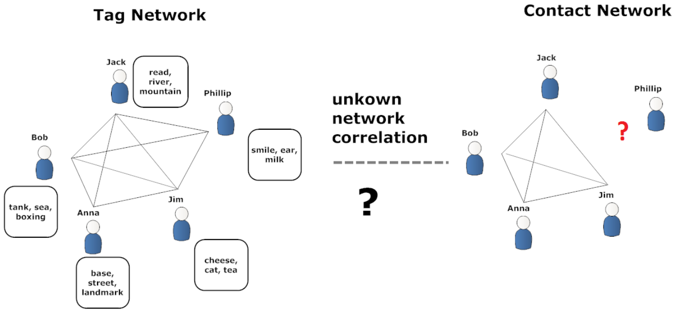

Fig. 1. Problem Illustration: how to correlate the two networks and recommend friends to Phillip

"Correlation" between networks means that the structures of different networks share some similar properties. Here the "structure" of a network is to some extent similar with "topology" but the meaning is broader: we define the structure of a network as the property of how the network is formed and organized. To determine the structure correlation, we propose to use the network alignment methods. It is defined as to find approximate isomorphisms between similar networks[8], and have been widely applied in the fields of bio-informatics[9] and computer vision[10]. In this paper we take advantage of the study about network alignment in other fields such as bio-informatics into social media as a new approach.

To model the network correlations, in this paper, we propose to align tag and contact networks through important tag feature selection. Here an "important" feature is decided by if a feature contributes much in correlating the tag network to 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 different social signals in different social networks, which may have different importance in mining the network correlations. 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. In Flickr network, he may find that most of his friends 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.

In addition to network alignment, to make more precise friend recommendation, we also consider network structure preservation in our algorithm. Here "preservation" means that we don't change much the tag network structure before and after alignment. By preserving 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[11]. In this paper we analyse its correctness mathematically and extend the idea to social media.

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

In summary, this paper makes the following contributions:

1. We have proposed 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 have developed a method by aligning these networks through important feature selection.

3. We also consider preserving the network structure for a more precise recommendation.

4. We have conducted 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 shows the performance of our method and analysis is made according to the result. Finally, 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 Social Network

Recommendation system have been widely used for different purpose such as item[12] or trip recommendation[13], media recommendation[14] and friend recommendation[15][16][17]. For friend recommendation, different methods and applications has been published: [15] considers user generated content on Twitter and develops a fast algorithm for real-time user-to-user similarity for follower/followee recommendation. It is developed only for text similarity. [16] concentrates on mobile applications and makes friend recommendation 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.

[17] also combines social information from different layers. It takes both context and content information and associates

them with domain knowledge. It takes user's implicit feedback for combination different kind of information. In addition to these works that combine different features such as location, time, text for similarity calculation, in this paper, we dig the network structure and select important features via network alignment to get a 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[18], bio-informatics[19], etc. However, algorithms run well on image and biological datasets need modifications for social media problems: 1). Social networks deal with large scale complex networks. 2). As discussed, different social networks are formed when people play different social roles. The correlations of these different social networks are not well studied. So unlike the case in image retrieval or bio-informatics, the network correlation in the field of social media has its own properties and requires further study. [20] is a pilot paper that studies the correlations among heterogeneous multimedia data. It studies the co-occurrence of the low-level features in different modalities and reinforce each other for information retrieval. But it doesn't study the correlation from a network view. [21] studies the matching of people's name and their social network identities such as their Twitter account with the help of common friends and co-occurrence of words. It illustrates that pure text similarity matching has poor matching performance. By synthetically considering the text, the popularity and the relationship among people, the correctly matching rate increases. This paper gives support to our idea that different social roles should be synthetically considered for a better recommendation. We further develop this idea in a way that we consider the structure of different networks and apply it for friend recommendation.

A more recent and related work for social media is given in [22]. In [22], first three networks are formed: user friendship network, tag network and image content network. Different relations are then defined within each network and between different networks. According to these relations the transition probability is defined and a random walk-based algorithm is developed to calculate the relative score among different nodes. This propagation algorithm can be used for multi-purpose recommendation such as item/query/friendship, because the links among different networks can be inferred. Compared with [22], our algorithm focuses on the use of new network alignment method. Though both of the two use multi networks for link prediction, our proposed algorithm correlates networks with same nodes and provides a mechanism to choose important features. It gives a new point of view to interpret the property of the social network, compared with the pure propagation algorithm in [22].

C. Network Alignment

To find the correlations between different social networks, we propose to use network alignment methods. Some previous researches consider the combination of different social networks for user behaviour prediction. [23] considers different behaviours such as music listening and booking reading for a composite behaviour prediction. It uses graphical model to build the relations of different networks. [24] utilizes the different application installation information from mobile devices. It would be better if these researches utilize the ample topological information of different network for 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 different fields that studies the relationship of big networks. In addition to the fields of bio-informatics and image processing problem as mentioned in introduction part, it has also been applied to different fields that deals with problem with large networks such as internet network management[25]. In [25], alignment is used to find the co-occurrence of elements of different networks and reduce the traffic of internet data. In this work, we extend network alignment concept to social network, by considering different social roles of individuals for a better recommendation.

D. Feature Selection

In this paper, we align different networks together through important feature selection. The initial motivation for feature selection is that the dimension of many social data is very high[26]. To deal with this problem we usually apply feature extraction[27] or feature selection[28] methods. Feature selection is often preferred over extraction, for the selected features have more understandable physical meanings. Feature selection has been successfully applied in the fields of biology and image processing[11]. In this paper we concentrate on unsupervised feature selection method.

[29] provides a clustering method based on spectral embedding. It projects the data on a subspace, chooses features that minimize the distance in each cluster on the projected subspace. However, it doesn't consider the pairwise structure of the original data. [11] provides a model that considers both the local and global structure preservation during projection. [11] induces the global preservation from linear kernel functions that can be applied for both supervised and unsupervised case. It is applied to image and bio datasets for clustering. Feature selection can also be applied in multimedia analysis that selects features from different domains[30]. In this paper we extend the traditional feature selection algorithm to the field of social media. By carefully analysis, we apply the concepts of structure preservation for a better use of social features for recommendation.

Some previous works have combined the concept of feature selection and similarity network(kernels) alignment together for different purpose. [31] studies the use of profile alignment and support vector machine for cellular localization. [32] has applied network alignment to overcome the problem that the locations of features varies from measurement to measurement for image matching. Our work differs from these previous works in two aspects: First, these works are mostly supervised and concentrate on image processing and bio-informatics. In our proposed network, we extend the concept to unsupervised cases and deal with more complex and bigger social network for social recommendation purpose. Second, most of the previous works are based on kernels that only applied the similarity information between users. Different from these kernel-based methods, we utilize more detailed information that is not only about the relationships between individuals, but also their social roles. These information is relatively easy to obtain in social media and so we expect better result 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 different networks including a contact network, $\mathcal{C}$ and $\mathcal{T}$(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 section, the different social roles of individuals are related to each other. $\mathcal{T}$ shows 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.

Specifically, 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 different 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 nations used in this paper. In total, there are N nodes in $\mathcal{C}$ and $\mathcal{T}$. The similarity matrix of 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 online 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 introduction part, we propose to apply feature selection techniques to find the alignment of different networks that have same nodes and different topologies.

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

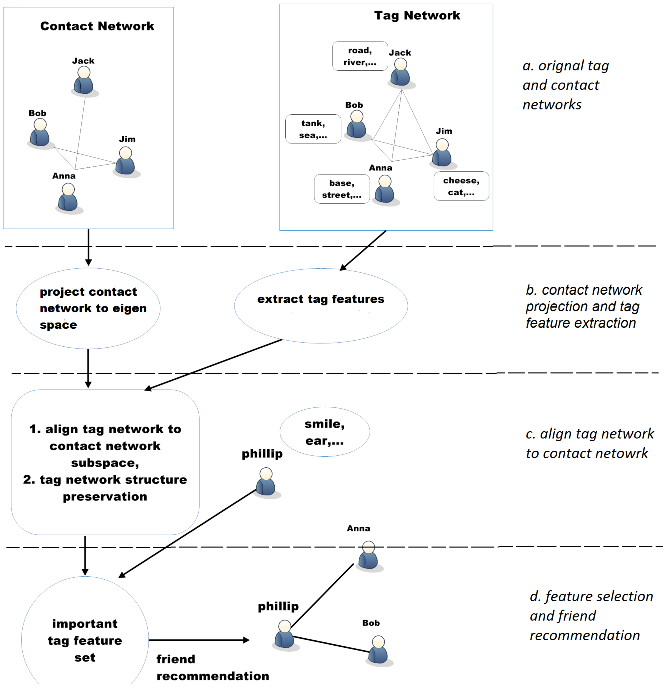
Contact Network
Jack
Bob
Jim
Anna
Tag Network
road, river,...
Jack
Bob
tank, sea,...
Jim
Anna
base, street,...
cheese, cat,...
a. orignal tag and contact networks
project contact network to eigen space
extract tag features
b. contact network projection and tag feature extraction
1. align tag network to contact network subspace,
2. tag network structure preservation
smile, ear,...
phillip
c. align tag network to contact netowrk
Anna
important tag feature set
phillip
Bob
friend recommendation
d. feature selection and friend recommendation

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 their subspace through feature selection. *d.* new friends will be recommended based on selected important features

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 to find the alignment of two networks. In this paper we align different networks by selecting important features that catch 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 few words. For example, when we find a friend on Facebook, usually we don't read all posts of a person, which is too time consuming. Instead we only read titles of several of his/her posts, and then we get a rough but to some extent accurate understanding of what he/she likes. Triggered by this phenomenon, we assume that individual's friend making decision is determined by a small amount 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 amount of features, different networks should be aligned well. 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. In this paper, we fix one network $\mathcal{C}$ and align the topologies of the other networks such as $\mathcal{T}$ in a subspace. Details for network correlation are given in 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 enough in the modified network. Thus the network alignment doesn't change the pairwise similarity among nodes. The effectiveness of pairwise similarity preservation has been shown in [11][33]. 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 it more carefully in IV-D.

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

B. Bag of Words and Feature Extraction

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

To calculate the tag feature matrix $\mathbf{X}$, we adopt the widely used TF-IDF method [34]. Except for counting the numbers of words that each user has used as the tags of his/her photos, TF-IDF assumes that seldom appeared features carry more information. Under this assumption, TF-IDF diminishes the weight of features that occur frequently in dataset and increases the weight of features that occur rarely. By calculating the TF-IDF of each word, we build feature matrix $\mathbf{X}$.

C. Network Alignment

As mentioned in IV-A, first we consider minimizing the gap between graph $\mathcal{C}$ 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:

$$\begin{aligned} \min_{\mathbf{W}} \parallel \mathbf{X}\mathbf{W}\mathbf{W}^T\mathbf{X}^T - \mathbf{K} \parallel_F^2 \qquad (1) \\ \text{subject to : } \mathbf{W} \in \{0,1\}^{F\times r} \\ \mathbf{W}^T\mathbf{1}_{F\times r} = \mathbf{1}_{r\times 1} \end{aligned}$$

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 $\mathbf{W}^T\mathbf{1}_{F\times r} = \mathbf{1}_{r\times 1}$ 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 we don't get a linear combination of features. This is a discrete problem and is hard to solve. Also, the user similarity in (1) measured by inner product makes the dimension too high for optimization. In [11], the problem is approximated by the following:

$$\min_{\mathbf{W}} \parallel \mathbf{XW} - \mathbf{V} \parallel_F^2 \qquad (2)$$

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 the data projects in. [11] shows that the upper bound of the solutions between eq.(2) and eq.(1) is small. By adding a regularization term we have the following optimization problem:

$$\min_{\mathbf{W}} \parallel \mathbf{XW} - \mathbf{V} \parallel_F^2 + \lambda \parallel \mathbf{W} \parallel_{2,p} \qquad (3)$$

The regularization term $\parallel \mathbf{W} \parallel_{2,p}$ forces $\mathbf{W}$ to be sparse. Traditional methods such as [11] often uses $l_{2,1}$ norm for sparse representation. Here we used $l_{2,p}$ norm following the ideas of [33]. By applying $l_{2,p}$ norm it is possible to control how sparse the projection matrix $\mathbf{W}$ should be.

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

D. Structure Preservation

Only considering gap minimization between two networks might lead to a problem that the structure of the modified network to be changed greatly. [11] have shown by experiments that by preserving the pairwise similarity of the original data structure, the clustering and classification performance is improved, compared with pure gap minimization. [11] 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 each two different nodes in the graph. For 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 number of co-occurrence features. However, this method is too simple and sometimes fails to catch the similarity among users. For example, a user with a tag "river" might have more similar topics with a user with "mountain" than another user with "basketball". The above mentioned simple method can't catch this similarity. In this paper we use Wordnet[35] to calculate the semantic similarity among different features[36]. The 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 [37], different methods can be applied for feature similarity measurements. Two different features will get a relation score between 0 and 1.

Recently there are other tag feature similarity methods in the field of natural language processing such as DSSM[38] and Google distance[39]. As Wordnet is widely applied in many previously works, we apply Wordnet in our work for similairity calculation by following the literature[35].

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

$$\mathbf{L}_{ij} = \sum_s \sum_t \mathbf{E}ij_{st} \tag{4}$$

2) Structure Preservation: After we represent the data structure, we study how to preserve the network structure during the alignment.

To minimize the pairwise distances, we define the data after feature selection as $\mathbf{A} = \mathbf{W}^T\mathbf{X}^T$, Also we set a pairwise distance matrix of all samples as $\mathbf{L}$, and then we want to minimize the in-dimension distance of all the r dimensions:

$$\begin{aligned}\min_{\mathbf{W}} \sum_{j=1}^{n}\sum_{i=1}^{n} \mathbf{A}_{1i}\mathbf{A}_{1j}\mathbf{L}_{ij} + \sum_{j=1}^{n}\sum_{i=1}^{n} \mathbf{A}_{2i}\mathbf{A}_{2j}\mathbf{L}_{ij} + \\ \cdots + \sum_{j=1}^{n}\sum_{i=1}^{n} \mathbf{A}_{ri}\mathbf{A}_{rj}\mathbf{L}_{ij}\end{aligned} \tag{5}$$

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

$$\begin{aligned} & \sum_{j=1}^{n}\sum_{i=1}^{n} \mathbf{A}_{1i}\mathbf{A}_{1j}\mathbf{L}_{ij} \qquad (6) \\ = \quad & \sum_{i=1}^{n} \mathbf{A}_{1i}\mathbf{L}_{i1}\mathbf{A}_{11} + \mathbf{A}_{1i}\mathbf{L}_{i2}\mathbf{A}_{12} + \cdots + \mathbf{A}_{1i}\mathbf{L}_{in}\mathbf{A}_{1n} \\ = \quad & [\sum_{i=1}^{n} \mathbf{A}_{1i}\mathbf{L}_{i1}, \sum_{i=1}^{n} \mathbf{A}_{1i}\mathbf{L}_{i2}, \cdots, \sum_{i=1}^{n} \mathbf{A}_{1i}\mathbf{L}_{in}](\mathbf{A}_{row1})^T \\ & = (\mathbf{AL})_{row1}(\mathbf{A}_{row1})^T = (\mathbf{ALA}^T)_{11} \end{aligned}$$

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

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

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

$$(5) = \mathrm{tr}(\mathbf{ALA}^T) = \mathrm{tr}(\mathbf{W}^T\mathbf{X}^T\mathbf{LXW}) \tag{8}$$

E. Solutions and Complexity Analysis

1) Solutions: By mixing up optimization problems in IV-C and IV-D together, we have the following optimization problems:

$$\min_{\mathbf{W}} \|\mathbf{XW} - \mathbf{V}\|_F^2 + \mu \mathrm{tr}(\mathbf{W}^T\mathbf{X}^T\mathbf{LXW}) + \lambda\|\mathbf{W}\|_{2,p} \tag{9}$$

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 [11], it can be approximated by $\mathbf{W}^T\mathbf{BW}$, where $\mathbf{B}$ is a diagonal matrix and defined as follows:

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

So equation (9) is formulated as follows:

$$\min_{\mathbf{W}} \|\mathbf{XW} - \mathbf{V}\|_F^2 + \mu \mathrm{tr}(\mathbf{W}^T\mathbf{X}^T\mathbf{LXW}) + \lambda \mathrm{tr}(\mathbf{W}^T\mathbf{BW}) \tag{11}$$

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 eq.(11) to zero, and then we get $\mathbf{W}$ as follows:

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

eq.(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. Notice that eq.(12) has the equivalent form as follows:

$$\mathbf{W} = \frac{1}{\lambda}\mathbf{B}^T(\frac{1}{\lambda}(\mathbf{I} + \mu\mathbf{L})\mathbf{XBX}^T + \mathbf{I})^{-1} \tag{13}$$

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

After getting $\mathbf{W}$, we update $\mathbf{B}$ by eq.(10). When p is no less than 1, it is a convex problem. According to [11], theoretically by iteratively updating $\mathbf{W}$ and $\mathbf{B}$, eq.(9) converges to a global optimum. For most cases in experiments, it converges to a reasonable small range within 5 rounds of iterations.

2) Feature Selection: The next is to choose important features according to the optimal $\mathbf{W}$. From eq.(1) we see that each row of $\mathbf{W}$ corresponds to one feature. The larger the norm of this row, the more important 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 nearly no contributions in alignment. So we rank the norm of rows of $\mathbf{W}$ according to their norms in a descending order, and choose features according to this order. The top features are considered to be most important for friend prediction. In the experimental part, we verify that by considering weight we get a slightly better performance.

3) Friend Recommendation: For a new user with some tag words coming into the network, how do we make recommendations according the important features? As mentioned before, we select these features based on the alignment from tag network to contact network, and these important features illustrate the correlations of contact and tag networks. So we calculate the similarities between tag features of the new user and those important features of the existing users. Because the important features pool reflects each existing user tag's contribution and the correlation between tag and contact networks, therefore, this similarity indicates the distance of the new user to those existing ones in the contact network. The more similar on the important tag set, the more closer the two uses should be. So by ranking the tag similarity of the new user and the members that already in the networks, we choose top K as recommending friends.

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 $\lambda$, $\mu$ 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: Initial $\mathbf{B}$ with identity matrix $\mathbf{I}$
 5: repeat
 6:     Calculate $\mathbf{W}$ by eq.(12) or eq.(13)
 7:     Calculate $\mathbf{B}$ by eq.(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: map the new user to the existing contact network by
    calculating the similarities between the important features
    of the new user and those of the existing users. This
    similarity indicates the distance of the new user to those
    existing ones in the contact network. So we choose the
    top $K$ users according to the similarity as recommended
    friends to the new user.
```

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

5) Complexity and Large Scale Data Suitability: The most time-consuming task in 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 calculating relation score matrix $\mathbf{E}$ is $F \times (F-1) \times \tau \,/\, 2$.

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

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

From the above discussion, it is clear that the complexity of the algorithm is mainly determined by $(\min\{N,F\}^3)$, which means that when the number of features is fixed, the complexity doesn't increase with the increase of the scale of data. So our algorithm fits large scale social network that contains millions of users. This is a good property of our algorithm.

V. Experiments

In this section we make extensive experiments to show the effectiveness of our proposed method, as well as illustrate some interesting properties. First we give a brief introduction of our social media data set, and then we discuss our algorithm from different aspects.

A. Experimental Settings

1) Dataset: We crawled a social network from the big image sharing site Flickr. To reduce bias, we crawled groups randomly. As the data set is quite large, a relatively unbias 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, in each group we crawled 5 users.

For each user we crawled all their photos, and tags of each photo. For same users, we crawled their contact information to form the contact network. In Flickr, the contact information is obtained by if a user has added another user to his/her friend list, or vise 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 :

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

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

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 $p \in [1, 1.5, 2, 2.5, 3]$.

We use the method stated in IV-E3 to recommend friend 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 usually when one becomes large, the other becomes small, so we use F-measure to combine the two:

$$F = 2 \times \frac{pecision \times recall}{precision + recall} \tag{14}$$

In our experiment we choose 80% for training and 20% for testing, and run totally 20 times to calculate the average precision, recall, and F-measure.

3) Reference Methods: We use several reference methods to show the advance of our proposed algorithm in friend recommendation. They are: 1). pure tag similarity, 2). SVM, 3). on-line collaborative filtering(OLCF)[40], and 4) Relational Domain Recommendation(RDR)[41].

The first is the simplest tag similarity comparison. We recommend friends of each user purely on the tag similarity calculated in IV-B.

The second method is an SVM-based method. It implements the "one-against-one" approach for multi-class classification[42]. We choose SVM as 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 5 users and we assume each group as one labelled class. Because each group is very small, we assume the members in each group are friends with each other. For training we choose 4 users in each group and test if the last one can be classified correctly. If it is correctly classified, we assume we have right friend recommendation result.

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

Some traditional model-based collaborative filtering methods face the problem that when a new user comes, the whole latent space has to be updated[12]. In this paper, we apply an online-updating collaborative filtering method as reference method[40].

At last we consider an multi-network based algorithm for comparison. When considering social multiple network problems, transition probability propagation is a method that is frequently used[41][22]. We choose [41] as a reference method for the following reasons: 1) It considers the relationships of different networks which is similar with our idea, though in [41], different networks have different nodes; 2). It uses the information of other networks for recommendation, which again has some similarities with ours. [41] enhances the links in one network and among different networks using a random walk propagation method. After enough round of walks it obtains the modified link weights between each user pair. And we use the weights for friend recommendation. [22] also uses a random walk base method but considers more kinds of relationships. Due to the limit of space we don't fulfil it here but leave it to further work.

B. *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 with certain number of features. We determine parameters in eq.(9) with a five cross validation procedure for the best value. The result of precision is shown in Figure 3:

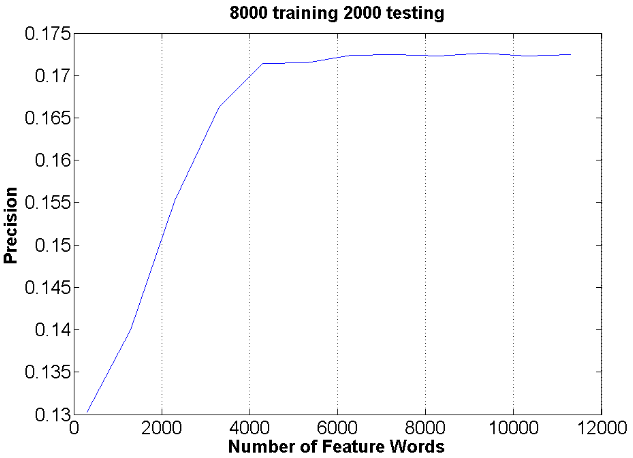

Fig. 3. Precision@20 with the Increase of Features

From Figure 3 we see that the algorithm performance 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 doesn't increase much. This means that when a user looks for friends, he/she only concentrates on some aspects of the characteristic of others but not all aspects.

C. *Feature Selection vs. Selection plus Re-weighting*

After the calculation of IV-E2 we get the importance of each features. Then we rank the features according their importance, and we get a list of features, on the top of the list are those features with most importance. Two methods can be applied in the next step: First, we choose important features from the top of the list; Second, we also choose the features from the top of the list, and consider their weightings for friend recommendation. The following are the results for pure feature selection and selection with re-weighting.

We fix the number of selected features to be 4500 and the total number of the dataset to be 10000, and change the number of recommended friends. The result is shown in figure 4:

We see that there is only slightly improvement when considering the weights of each feature.

D. *Comparison of Different Methods for Precision*

In this experiment we compare the proposed method with all reference methods mentioned in section V-A3.

We fix the number of features for proposed method to be 4500. Now we compare the Precision Measure performance

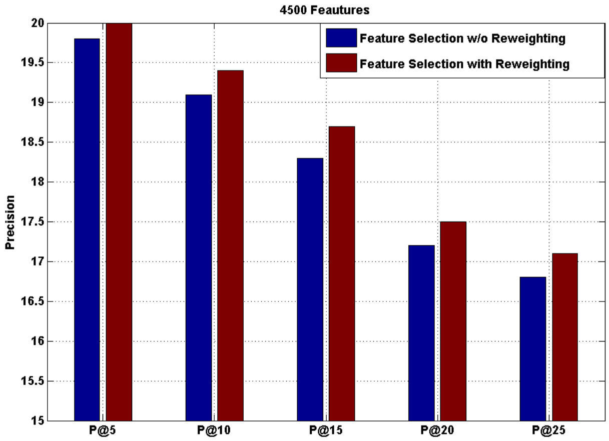

Fig. 4. Feature Selection vs. Re-weighting

of different methods with the reference methods mentioned in V-A3. We change the value of K from 5 to 30 for a more complete comparison. The resulting histogram is given in Figure 5: From Figure 5 we see that no matter which K we choose, the proposed method outperforms all other reference methods. Pure similarity method has the lowest recommendation precision. This coincide our statement in Introduction, that people make friends not only based on similarity.

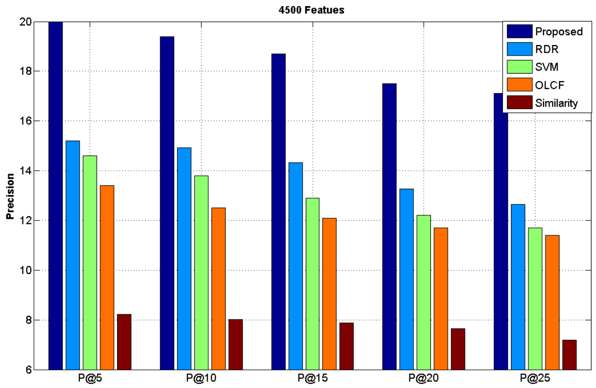

Fig. 5. Precision with Different Algorithms for 4500 Features

Collaborative filtering method OLCF has slightly lower performance than SVM. The reason might be that by filling the user-feature matrix, it adds some noise and redundancy. So it doesn't have a good performance.

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 than SVM. However, it lacks a mechanism to consider what is important in 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 those most important features when people make friends with each other.

When we increase the number of selected feature to be 7500, the result is shown in Figure 6

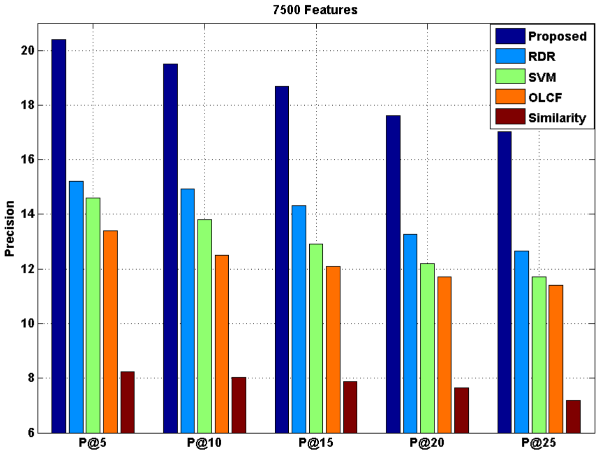

Fig. 6. Precision with Different Algorithms for 7500 Features

Figure 6 shows that the performance of proposed method does not increase much with increase of features, when the number of features is relatively large. The reason is that our weighing matrix $\mathbf{W}$ is sparse, and for features that have a low order, their weights is very small or even becomes zero. So they don't have much influence on the recommendation accuracy.

E. Comparison for Recall and F-measure

As mentioned before, recall and F-measure are also common metrics to measure the effectiveness of recommendation algorithms. In the following, Figure 7 and Figure 8 give the results of our proposed method and the reference methods. Here we fix the number of recommended users to be 5, the number of features to be 4500, and change the total number of users from 5000 to 10000.

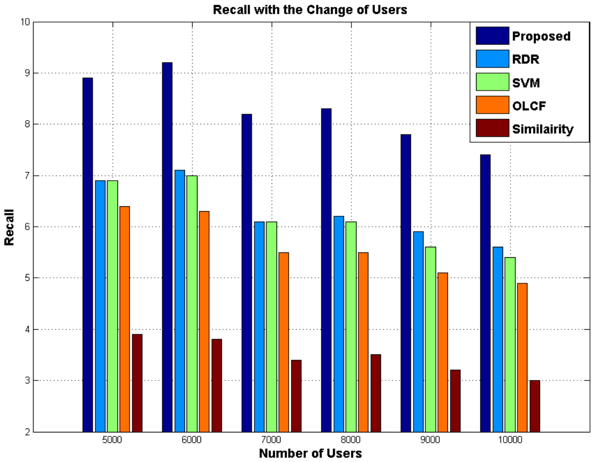

Fig. 7. Recall with Different Algorithms

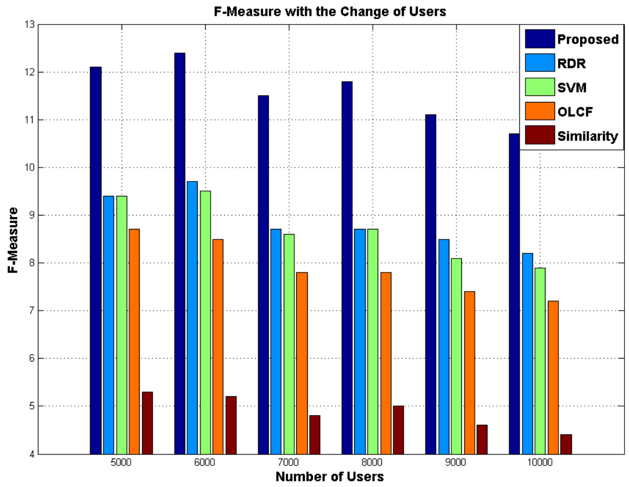

Fig. 8. F-Measure with Different Algorithms

From Figure 7 and 8 we see that the system performance changes when the number of users are changed. And our proposed method always has the best performances.

F. Effect of Parameters

In this experiment we study the effect of parameter μ. It adjusts the weight between alignment of two networks and preservation of data structure. Still we fix the feature number to be 4500 and K to be 20, λ is fixed to be 1. The result is given in figure 9:

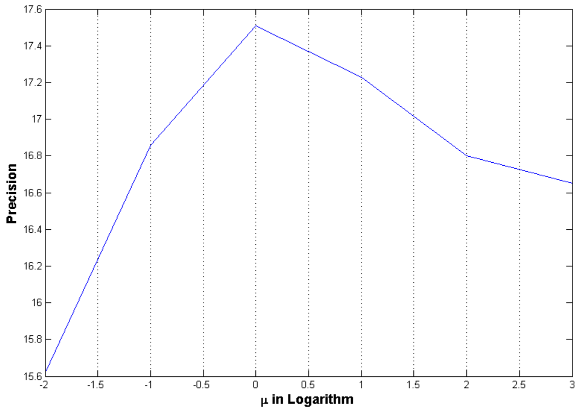

Fig. 9. Effect of weights between network alignment and structure preservation

Figure 9 shows that with the increase of μ, first the prediction accuracy increase to a maximum point, then it decreases. This phenomenon tells us that both the network alignment and network structure preservation play important role in important feature selection. A maximum friend recommendation precision is reached when we balance both of the two well.

Different choice of p shows how sparse the projection matrix $\mathbf{W}$ is. the following is the experimental result.

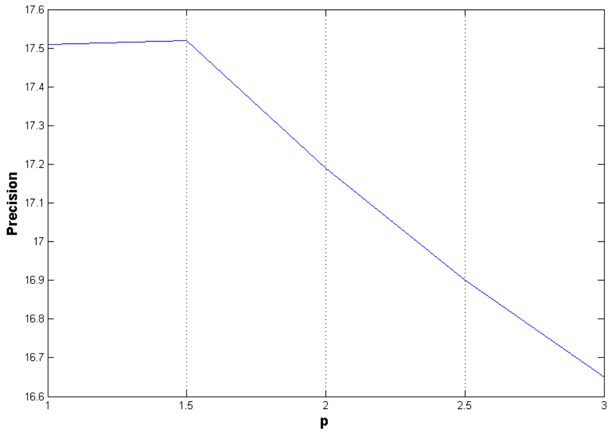

Fig. 10. Effect of Sparsity of $\mathbf{W}$

Figure 10 tells us that the sparsity of $\mathbf{W}$ does have influence on friend recommendation accuracy. As p increases, the recommendation accuracy first keeps nearly unchanged, and then goes down quickly. The reason might be that as $\mathbf{W}$ becomes more dense, it lacks the ability to distinguish the most important features from others. As p doesn't influence much from 1 to 1.5, for simplicity reasons, in other parts of our experiments we fix p to be 1.

VI. Conclusions and Further Works

In this paper, we study the friend recommendation problem from the view of network correlation. A person has many different social roles on-line. For 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 among different networks. The second aspect we take into account is the 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: we achieve the highest precision in friend prediction. We found that a small number of features can align the tag network to contact network well and 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, And 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 correlations 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 *ACM CHI'09*, New York, NY, USA, April 2009.

[2] S. Wan, Y. Lan, J. Guo, C. Fan, and X. Cheng, "Informational friendship recommendation in social media," in *ACM SIGIR'13*, Dublin, Ireland, pp. 1045–1048, July 2013.

[3] E. Barnett and M. Casper, "A defination of 'social environment'," *American Journal of Public Health*, vol. 91, March 2001.

[4] R. J. Rummel, *The Conflict Helix:Principles and Practices of Interpersonal, Social, and International Conflict and Cooperation*. New Brunswick, N.J, 1991.

[5] M. Weber, *The Nature of Social Action*. Cambridge University Press, 1991.

[6] R. Brym, *Sociology: Your Compass for a New World, Brief Edition*. 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 *ACM KDD'13*, pp. 695–703, August 2013.

[8] M. Bayati, M. Gerritsen, D. Gleich, A. Saberi, and Y. Wang, "Algorithms for large, sparse network alignment problems," in *ICDM'09*, Miami, FL ,USA, pp. 705–710, Dec 2009.

[9] G. Klau, "A new graph-based method for pairwise global network alignment," *BMC Bioinformatics*, vol. 10, p. S59, January 2009.

[10] D. Conte, P. P. Foggia, C. Sansone, and M. Vento, "Thirty years of graph matching in pattern recognition," *Intl. J. of Pattern Recognition and Artificial Intelligence*, vol. 18, pp. 265–298, May 2004.

[11] X. Liu, L. Wang, J. Zhang, J. Yin, and H. Liu, "Global and local structure preservation for feature selection," *IEEE Transaction on Neural network and learning system*, pp. 1083–1095, November 2014.

[12] H. Ma, H. Yang, M. R. Lyu, and I. King, "Sorec: social recommendation using probabilistic matrix factorization," in *ACM CIKM'08*, Napa Valley, CA, USA, pp. 931–940, November 2008.

[13] Y.-Y. Chen, A.-J. Cheng, and W. H. Hsu, "Travel recommendation by mining people attributes and travel group types from community-contirbuted photos," *IEEE Trans. on Multimedia*, vol. 15, pp. 1283–1295, October 2013.

[14] S. D. Roy, T. Mei, W. Zeng, and S. Li, "Socialtransfer: Cross-domain transfer learning from social streams for media applications," in *ACM MM'12*, Nara, Japan, pp. 649–658, Octorber 2012.

[15] J. Hannon, M. Bennett, and B. Smyth, "Recommending twitter users to follow using content and collaborative filtering approaches," in *ACM RecSys'10*, Barcelona, Spain, pp. 199–206, September 2010.

[16] N. Li and G. Chen, "Multi-layered friendship modeling for location-based mobile social networks," in *IEEE MobiQuitous'09*, Toronto, Canada, pp. 1–10, July 2009.

[17] X. Xie, "Potential friend recommendation in online social networking," in *IEEE/ACM CPSCom'10*, Hangzhou, China, pp. 831–835, December 2010.

[18] M. Gori, M. Maggini, and L. Sarti, "Exact and approximate graph matching using random walks," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 27, pp. 1100–1111, July 2005.

[19] M. H. Goodfellow, "Biochemical network matchng and composition," in *EDBT'10*, Lausanne, Swizerland, March 2010.

[20] Y.-T. Zhuang, Y. Yang, and F. Wu, "Mining semantic correlation of heterogeneous multimedia data for cross-media retrieval," *IEEE Trans. on Multimedia*, vol. 10, pp. 221–229, February 2008.

[21] G. won You, S. won Hwang, Z. Nie, and J.-R. Wen, "Socialsearch: enhancing entity search with social network matching," in *EDBT'11*, Upplsala, Sweden, pp. 515–519, March 2011.

[22] D. Liu, G. Ye, C.-T. Chen, S. Yan, and S.-F. Chang, "Hybrid social media network," in *ACM MM'12*, Nara,Japan, pp. 659–668, October 2012.

[23] E. Zhong, W. Fan, and Q. Yang, "User behavior learning and transfer in composite social networks," vol. 8, February 2014.

[24] W. Pan, N. Aharony, and A. S. Pentland, "Composite social network for predicting mobile apps installation," in *AAAI'11*, San Francisco, CA, USA, August 2011.

[25] C. Kreibirch and J. Crowcroft, "Efficient sequence alignment of network traffic," in *ACM IMC'06*, Rio de Janeiro, Brazil, pp. 307–312, October 2006.

[26] R. E. Bellman, *Dynamic Programming*. Courier Dover Publications, 2003.

[27] I. T. Jolliffe, *Principal component analysis*. New York: Springer, 1986.

[28] H. Liu and H. Mototda, *Computational methods for feature selection*. Chapman & Hall, 2008.

[29] D. Cai, C. Xhang, and X. He, "Unsupervised feature selection for multi-cluster data," in *ACM SIGKDD'10*, Washington DC, USA, pp. 333–342, July 2010.

[30] Y. Yang, Z. Ma, A. G. Hauptmann, and N. Sebe, "Feature selection for multimedia analysis by sharing information among multiple tasks," *IEEE Trans. on Multimedia*, vol. 15, pp. 661–669, 2013.

[31] J. Guo, M. Man-Wai, and S.-Y. Kung, "Eukaryotic protein subcellular localization based on local pairwise profile alignment svm," in *IEEE 16th MLSP Workshop*, pp. 391–396, September 2006.

[32] P. Kosir, R. DeWall, and R. A. Mitchell, "A multiple measurement approach for feature alignment," in *IEEE NAECON'96*, vol. 1, pp. 94–101, May 1996.

[33] Y. Yang, Z. Xu, G. Liu, Z. Ma, and N. Sebe, "Global structual feature selection with sparsity for multimedia data understanding," in *ACM MM'13*, pp. 537–540, October 2013.

[34] 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. Information SYstems*, vol. 26, June 2008.

[35] http://wordnet.princeton.edu/.

[36] P. Nithiya, V. Vidhya, and L. Ganesan, "Development of semantic based information retrieval using word-net approach," in *ICCCNT'10*, Karur, India, pp. 320–323, April 2010.

[37] Y. Jin, L. Khan, L. Wang, and M. Awad, "Image annotations by combining multiple evidence & wordnet," in *ACM MM'05*, vol. 15, Singapore, pp. 706–715, November 2005.

[38] https://microsoft.sharepoint.com/teams/DSSM_Text_Processing.

[39] R. Cilibrasi and P. M. B. Vitanyi, "The google similarity distance," *IEEE Trans. Knowledge and Data Engineering*, vol. 19, no. 3, pp. 370–383, 2007.

[40] S. Rendle and L. S. Thieme, "Online-updating regularized kernel matrix factorization models for large-scale recommender systems," in *ACM RecSys'08*, Lausanne, Swizerland, pp. 251–258, Octorber 2008.

[41] M. Jiang, P. Cui, F. Wang, Q. Yang, W. Zhu, and S. Yang, "Social recommendation across multiple relational domains," in *ACM CIKM'12*, Maui, HI, USA, October 2012.

[42] S. Knerr, L. Personnaz, and G. Dreyfus, "Single-layer learning revisited: A stepwise procedure for building and training a neural network," *Neurocomputing: Algorithms, Architectures and Application*, vol. F68, pp. 41–50, 1990.