Inferring Anchor Links across Multiple Heterogeneous Social Networks

Xiangnan Kong
University of Illinois at Chicago
Chicago, IL, USA
xkong4@uic.edu

Jiawei Zhang
University of Illinois at Chicago
Chicago, IL, USA
jzhan9@uic.edu

Philip S. Yu
University of Illinois at Chicago
Chicago, IL, USA
King Abdulaziz University
Jeddah, Saudi Arabia
psyu@cs.uic.edu

ABSTRACT

Online social networks can often be represented as heterogeneous information networks containing abundant information about: who, where, when and what. Nowadays, people are usually involved in multiple social networks simultaneously. The multiple accounts of the same user in different networks are mostly isolated from each other without any connection between them. Discovering the correspondence of these accounts across multiple social networks is a crucial prerequisite for many interesting inter-network applications, such as link recommendation and community analysis using information from multiple networks. In this paper, we study the problem of anchor link prediction across multiple heterogeneous social networks, *i.e.*, discovering the correspondence among different accounts of the same user. Unlike most prior work on link prediction and network alignment, we assume that the anchor links are one-to-one relationships (*i.e.*, no two edges share a common endpoint) between the accounts in two social networks, and a small number of anchor links are known beforehand. We propose to extract heterogeneous features from multiple heterogeneous networks for anchor link prediction, including user's social, spatial, temporal and text information. Then we formulate the inference problem for anchor links as a stable matching problem between the two sets of user accounts in two different networks. An effective solution, MNA (Multi-Network Anchoring), is derived to infer anchor links *w.r.t.* the one-to-one constraint. Extensive experiments on two real-world heterogeneous social networks show that our MNA model consistently outperform other commonly-used baselines on anchor link prediction.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications-Data Mining

Keywords

Heterogeneous Social Network, Multi-Network, Anchor Links

CIKM'13, Oct. 27–Nov. 1, 2013, San Francisco, CA, USA.

http://dx.doi.org/10.1145/2505515.2505531.

1. INTRODUCTION

Online social networks, such as Facebook, Twitter and Foursquare[1], have become more and more popular in recent years. Each social network can often be represented as a heterogeneous network containing abundant information about: who, where, when and what. Nowadays, people are getting involved in more and more different kinds of social networks simultaneously. For example, people usually share reviews or tips about different locations or places with their friends using Foursquare network. At the same time, they may also share the latest news using Twitter network, and share photos using Facebook network. Thus, each user often has multiple separate accounts in different social networks. However, these accounts of the same user are mostly isolated without any connection or correspondence to each other.

Discovering the correspondence between accounts of the same user is a crucial prerequisite for many interesting inter-network applications, such as link recommendation and community analysis using information from multiple networks. For example, in Foursquare network, the social connections and activities of new users can be very sparse. The friend and location recommendations for such users are very hard using only one network. However, if we also know the user's Twitter account, his/her social connections and location data in Twitter network can also be used to improve the recommendation performances in the Foursquare network.

Figure 1 shows an example of two heterogeneous social networks (Twitter and Foursquare) with six users. Each user has two accounts in two networks separately. In each network, users are connected with each other through social links. Moreover, each user is also connected with a set of locations, timestamps and text contents through online activities. Note that the top two users in Figure 1 also have another type of link, which connects the same user's accounts in two networks. We call these links as **anchor links**. Each anchor link indicates a pair of accounts that belong to the same user. The task of anchor link prediction is to discover which *pair* of accounts, as shown with question marks in Figure 1, belong to the same user in real-world.

The problem of anchor link prediction across multiple heterogeneous social networks has not been studied in this context so far. Unlike most prior work on link prediction [9, 13, 10, 23, 14] and network alignment [3], we assume that anchor links are one-to-one relationships among the two sets of user accounts (*i.e.*, no two edges share a common end-

[1]https://foursquare.com

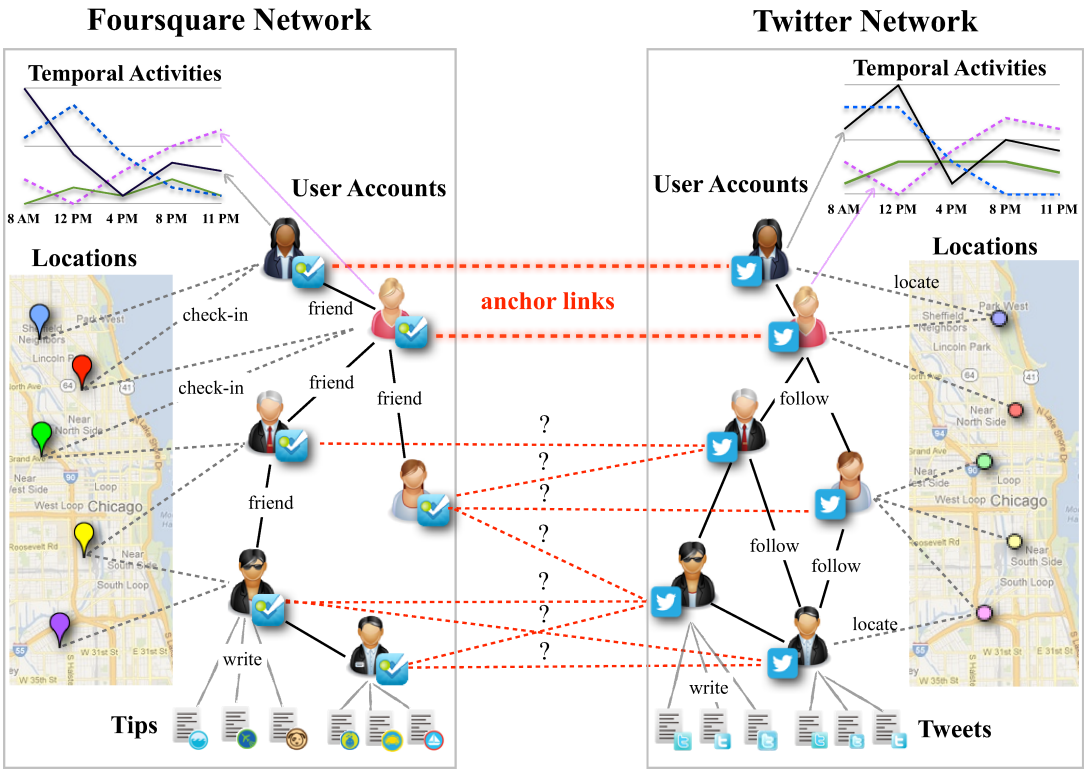


Figure 1: Example of inferring anchor links across two heterogeneous social networks: Foursquare network and Twitter network.

point[2]) and a small number of anchor links across networks are known beforehand. A detailed comparison between anchor link prediction problem and other related problems is shown in Table 1.

Despite its value and significance, the anchor link prediction task across multiple heterogeneous social network is very challenging due to the specific characteristics of the task. The reasons are listed as follows.

- *Lack of features.* Most existing features for link prediction, such as *common neighbors* and *Jaccard's coefficient*, apply only in single network settings. In order to compute these features, the target links are required to be many-to-many relationships among a set of nodes in one single network. However, in anchor link prediction problem, the anchor links are one-to-one relationships across multiple networks. Existing features in link prediction will reduce to a constant value, if we directly apply them on anchor link prediction problem.

- *Inference w.r.t. constraints.* Another fundamental problem in anchor link prediction lies in the one-to-one constraint in the inference step. Conventional supervised link prediction approaches usually assume that the target links to predict are many-to-many relationships. Thus they cannot be directly used in anchor link prediction problem, since the one-to-one constraint may not hold during the inference process. Note that in anchor link prediction tasks, the labels of different candidate anchor links are correlated and should be predicted collectively due to one-to-one constraint.

- *Uncalibrated scores.* Conventional supervised link prediction methods can usually predict a ranking score for each pair of nodes. However, these predicted scores are uncalibrated in scale for anchor link prediction tasks. In order to make accurate anchor link predictions, we need to calibrate these scores in a meaningful way to facilitate the inference process.

In this paper, we introduce a novel framework to tackle the above issues. Different from existing link prediction methods, our approach, called MNA (Multi-Network Anchoring), can extract heterogeneous features from multiple heterogeneous networks for anchor link prediction, including user's social, spatial, temporal and text information. We extended some existing social features for link prediction into multi-network settings, based upon the known anchor links. Then we train a binary classifier on the training set for anchor link prediction. In the inference step, we propose to formulate the anchor link inference problem as a stable matching problem based upon the scores of the binary classifier. MNA method can effectively infer the anchor links w.r.t. one-to-one constraint. We run extensive experiments on real-world heterogeneous social networks. The results show that our MNA model consistently outperforms other commonly-used baselines.

[2]We ignore the case that an individual can have multiple accounts in the same network which is a different problem [4].

Table 1: Summary of related problems.

Property	**Inferring Anchor Links**	**Link Prediction** [9, 13, 10, 23, 14]	**Network Alignment** [3]	**Relational Entity Resolution** [4]
target relationship	one-to-one	many-to-many	one-to-one	clustering
network	heterogeneous	homogeneous/heterogeneous	homogeneous	homogeneous/heterogeneous
#network	multiple	single/multiple	multiple	single
setting	supervised	supervised/unsupervised	unsupervised	unsupervised
target link type	inter-network	intra-network	inter-network	intra-network

The rest of the paper is organized as follows. We first introduce the preliminary concepts, give the problem analysis in Section 2. In Section 3, we propose the MNA method for anchor link prediction across multiple networks. Then Section 4 reports the experiment results on real-world social networks. In Section 6, we conclude the paper.

2. PROBLEM FORMULATION

In this paper, we focus on studying the anchor link prediction problem on two heterogeneous social networks, though the proposed framework can easily be generalized to the settings with more than two networks.

Suppose we are given a source network $\mathcal{G}^s$ and a target network $\mathcal{G}^t$, which are both heterogeneous social networks. Formally, we represent each heterogeneous social network as an undirected graph. The source network $\mathcal{G}^s = (\mathcal{V}^s, \mathcal{E}^s)$ contains different types of nodes and links. $\mathcal{V}^s = \mathcal{U}^s \cup \mathcal{L} \cup \mathcal{T} \cup \mathcal{W}$ is the set of nodes in the source network, which includes four types of nodes. $\mathcal{U}^s = \{u_1^s, u_2^s, \cdots, u_N^s\}$ is the set of user accounts. $\mathcal{L} = \{\ell_1, \ell_2, \cdots, \ell_L\}$ is the set of L different locations or places, where users have published their posts at. $\mathcal{T} = \{t_1, t_2, \cdots\}$ represents a set of time slots that users have published posts at. Each time slot can be an hour of a day, or a day in a month. $\mathcal{W} = \{w_1, w_2, \cdots\}$ is the set of words that users have used in their posts. $\mathcal{E}^s \subset \mathcal{V}^s \times \mathcal{V}^s$ is the edges of different types in the heterogeneous social network $\mathcal{G}^s$. $\Gamma_s \subset \mathcal{E}^s$ is the set of user pairs that are friends with each other in network $\mathcal{G}^s$.

Similarly, we define the target network as $\mathcal{G}^t = (\mathcal{V}^t, \mathcal{E}^t)$. $\mathcal{U}^t$ denotes the set of user accounts in the target network. Without loss of generality, we assume the source and target network share the same sets of locations $\mathcal{L}$, time slots $\mathcal{T}$ and words $\mathcal{W}$.

Anchor Link Prediction: Suppose we have two heterogeneous social networks $\mathcal{G}^s$ and $\mathcal{G}^t$, with a small set of known anchor links between the users accounts in two networks, $\mathcal{A} = \{(u_i^s, u_j^t), u_i^s \in \mathcal{U}^s, u_j^t \in \mathcal{U}^t\}$. Anchor links are one-to-one relationships between user accounts in $\mathcal{U}^s$ and $\mathcal{U}^t$, *i.e.*, no two anchor links share a same user account. (u_i^s, u_j^t) denotes that the two user accounts belong to the same user. The task of anchor link prediction is to predict whether there is an anchor link between a pair of user accounts u_i^s and u_j^t, where $u_i^s \in \mathcal{U}^s, u_j^t \in \mathcal{U}^t$.

The key issue of *anchor link prediction* is to learn a one-to-one matching between the user accounts of two heterogeneous social networks. This problem formulation is different from existing works on social link prediction [9, 13, 10, 23, 14] mainly in two-folds: First, the target links to predict are one-to-one relationships between two sets of nodes, *e.g.*, Twitter accounts and Facebook accounts. How can we extract informative features for anchor link prediction task? Existing features for link prediction, such as number of common neighbors and the shortest distance, require that the target links should be many-to-many relationships. Second, the prediction of all anchor links should be considered collectively due to the one-to-one constraint. Supervised link prediction methods usually make predictions on a set of links independently, because there is no constraint on the degree of each node in the network.

3. MULTI-NETWORK ANCHORING

We design a two-phase approach to address the major challenges of anchor link prediction. The first phase tackles feature extraction problem, while the second phase takes care of one-to-one constrained anchor link prediction. The phase of feature extraction mainly explore two kinds of ideas on multiple heterogeneous social networks. First, we exploit social links in each network and the labeled anchor links across the two networks to extract *social features* for anchor link prediction. Second, we exploit the heterogeneous information in both networks to extract three sets of heterogeneous features for anchor link prediction, which correspond to aggregated patterns of users on Spatial distribution, temporal activity distribution and text content distribution separately. We use all the extracted features and the pairs of accounts with known labels to learn a binary SVM for anchor link prediction. Since the label predictions of SVM don't satisfy the one-to-one constraint, we use real-value scores of the SVM as the input for the second phase, and derive the anchor link predictions collectively according to the one-to-one constraint.

3.1 Extracting Heterogeneous Features across Networks

Most existing features for link prediction, such as number of common neighbors, focus on single network settings, and the target links are assumed to be many-to-many relationships. These features cannot be directly used in anchor link prediction across multiple networks.

3.1.1 Multi-Network Social Features

Users often have similar social links in different social networks, such as Twitter and Facebook, because such social links usually indicate the user's social ties in real life. We can make use of the social similarity between two user accounts from different social networks to help locate the same user.

Our goal is to extract discriminative social features for a pair of user accounts in two disjoint social networks. Intuitively, the social neighbors of each user account can only involve user accounts from the same social network. For example, the neighbors for a Facebook account can only involve Facebook accounts instead of Twitter accounts. However, in anchor link prediction problem, we need to extract a

set of features about a pair of user accounts in two different networks separately. The social neighbors for two user accounts are two disjoint sets of user accounts in two separate networks. There can not exist any shared nodes among the neighbours of the pair of user accounts. In the following, we propose to extend several social features to multi-network settings.

Here we extend the definition of some commonly used social features in link prediction, *i.e.*, "*common neighbors*", "*Jaccard's coefficient*" and "*Adamic/Adar measure*" [1].

• **Extended Common Neighbors**: $CN(u_i^s, u_j^t)$ represents the number of 'common' neighbors between u_i^s in the source network and u_j^t in the target network. We denote the neighbors of u_i^s in the source network as$\Gamma_s(u_i^s)$, and the neighbors of u_j^t in the target network as$\Gamma_t(u_i^t)$. We define the measure of *extended common neighor* as the number of known anchor links between$\Gamma_s(u_i^s)$ and$\Gamma_t(u_i^t)$.

$$\begin{aligned} CN(u_i^s, u_j^t) &= \left|\{(u_p^s, u_q^t) \in \mathcal{A}, u_p^s \in \Gamma_s(u_i^s), u_q^t \in \Gamma_t(u_j^t)\}\right| \\ &= \left|\Gamma_s(u_i^s) \bigcap_{\mathcal{A}} \Gamma_t(u_j^t)\right| \end{aligned}$$

It indicates how many pairs of user accounts belong to a same user.

• **Extended Jaccard's coefficient**: We can extend the measure of Jaccard's coefficient to multi-network setting using similar method of extending common neighbor. $JC(u_i^s, u_j^t)$ is a normalized version of common neighbors, *i.e.*, $CN(u_i^s, u_j^t)$ divided by the total number of distinct users in$\Gamma_s(u_i^s) \cup \Gamma_t(u_j^t)$:

$$JC(u_i^s, u_j^t) = \frac{\left|\Gamma_s(u_i^s) \bigcap_{\mathcal{A}} \Gamma_t(u_j^t)\right|}{\left|\Gamma_s(u_i^s) \bigcup_{\mathcal{A}} \Gamma_t(u_j^t)\right|}$$

where

$$\left|\Gamma_s(u_i^s) \bigcup_{\mathcal{A}} \Gamma_t(u_j^t)\right| = |\Gamma_s(u_i^s)| + |\Gamma_t(u_j^t)| - \left|\Gamma_s(u_i^s) \bigcap_{\mathcal{A}} \Gamma_t(u_j^t)\right|$$

• **Extended Adamic/Adar Measure**: Similarly, we also extend the Adamic/Adar Measure into multi-network settings, where the common neighbors are weighted by their average degrees in both social networks.

$$AA(u_i^s, u_j^t) = \sum_{\forall (u_p^s, u_q^t) \in \Gamma_s(u_i^s) \bigcap_{\mathcal{A}} \Gamma_t(u_j^t)} \log^{-1}\left(\frac{|\Gamma_s(u_p^s)| + |\Gamma_t(u_q^t)|}{2}\right)$$

3.1.2 Heterogeneous Features across Networks

In addition to the social features mentioned above, heterogeneous social networks also involve abundant information about: where, when and what. In the following, we propose to exploit the spatial, temporal and text content information about different user accounts to facilitate anchor link prediction.

• **Spatial distribution features:** We notice that users in different social networks usually publish posts at similar locations in real-life, such as their home, working places, traveling spots, *etc.* We can make use of the similarity between the spatial distributions of two user accounts from different social networks to help locate the same user. Each location can be represented as a pair of (latitude, longitude) $= \ell \in \mathcal{L}$.

Algorithm 1 Multi-Network Anchoring

Input: two heterogeneous social networks, $\mathcal{G}^s$ and $\mathcal{G}^t$.
a set of known anchor links $\mathcal{A}$
Output: a set of inferred anchor links $\mathcal{A}'$

```
Construct a training set of user account pairs with known labels using A.
For each pair (u_i^s, u_j^t), extract four types of features.
Training classification model C on the training set.
Perform classification using model C on the test set.
For each unlabeled user account, sort the ranking scores into a preference list of the matching accounts.
Initialize all unlabeled u_i^s in G^s and u_j^t in G^t as free
A' = ∅
while ∃ free u_i^s in G^s and u_i^s's preference list is non-empty do
    Remove the top-ranked account u_j^t from u_i^s's preference list
    if u_j^t is free then
        A' = A' ∪ {(u_i^s, u_j^t)}
        Set u_i^s and u_j^t as occupied
    else
        ∃u_p^s that u_j^t is occupied with.
        if u_j^t prefers u_i^s to u_p^s then
            A' = (A' − {(u_p^s, u_j^t)}) ∪ {(u_i^s, u_j^t)}
            Set u_p^s as free and u_i^s as occupied
        end if
    end if
end while
```

We propose to use three measures to evaluate the similarity between the spatial distributions of two users accounts: 1) the number of shared locations; 2) the cosine similarity between the two sets of locations; 3) the average distance between the two sets of locations.

• **Temporal distribution features:** We also notice that users in different social networks usually publish posts at similar time slots in real-life, such as hours after work and weekends, *etc.* Such temporal distribution indicates the user's online activity patterns. For example, some users may like to send tweets at night, while other users may like to write tweets at commuting time on the bus or train. The temporal distribution of different user accounts can also help us find the anchor links between two networks. We extract similar measures about the spatial distributions for two user accounts: 1) the number of shared time slots when publishing posts; 2) the cosine similarity between the two vectors of temporal activities.

• **Text content features:** The text content of posts by users in different social networks can also hint for the anchor links, because different users may have different choices of words in their posts. We first convert the posts of each user account into a bag-of-words vector weighted by TF-IDF. Then for each pair user accounts, we compute two kinds of similarities as features: 1) the inner product of the two vectors; 2) the cosine similarity of the two vectors.

3.2 Inferring anchor links w.r.t. one-to-one constraints

After extracting all the four types of heterogeneous features in the previous section, we can train a binary classifier, such as SVM or logistic regression, for anchor link prediction. However, in the inference process, the predictions of

the binary classifier cannot be directly used as anchor links due to the following issues:

- The inference of conventional classifiers are designed for constraint-free settings, and the one-to-one constraint may not necessarily hold in the label prediction of the classifier (SVM).
- Most classifiers also produce output scores, which can be used to rank the data points in the test set. However, these ranking scores are uncalibrated in scale to anchor link prediction task. Previous classifier calibration methods [26] apply only to classification problems without any constraint.

In order to tackle the above issues, we propose an inference process, called Mna (Multi-Network Anchoring), to infer anchor links based upon the ranking scores of the classifier. Our solution is motivated by the *stable marriage problem* [8] in mathematics.

We first use a toy example in Figure 2 to illustrate the main idea of our solution. Suppose in Figure 2(a) we are given the ranking scores from the classifiers. We can see in Figure 2(b) that link prediction methods with a fixed threshold may not be able to predict well, because the predicted links do not satisfy the constraint of one-to-one relationship. Thus one user account in the source network can be linked with multiple accounts in the target network. In Figure 2(c), *weighted maximium matching* methods can find a set of links with maximum sum of weights. However, it is worth noting that the input scores are uncalibrated, so maximum weight matching may not be a good solution for anchor link prediction problems. The input scores only indicate the ranking of different user pairs, *i.e.*, the preference relationship among different user pairs.

Here we say 'node x prefers node y over node z', if the score of pair (x, y) is larger than the score of pair (x, z). For example, in Figure 2(c), the weight of pair a, *i.e.*, Score$(a) = 0.8$, is larger than Score$(c) = 0.6$. It shows that user u_1^s (the first user in the source network) *prefers* u_1^t over u_2^t. The problem with the prediction result in Figure 2(c) is that, the pair (u_1^s, u_1^t) should be more likely to be an anchor link due to the following reasons: (1) u_1^s prefers u_1^t over u_2^t; (2) u_1^t also prefers u_1^s over u_2^s.

Definition (Blocking Pair): A pair (u_i^s, u_j^t) is a blocking pair iff u_i^s and u_j^t both prefer each other over their current assignments respectively in the predicted set of anchor links $\mathcal{A}'$.

Definition (Stable Matching): An inferred anchor link set $\mathcal{A}'$ is stable if there is no blocking pair.

We propose to formulate the anchor link prediction problem as a stable matching problem between user accounts in source network and accounts in target network. Assume that we have two sets of unlabeled user accounts, *i.e.*, $\mathcal{U}^s = \{u_i^s\}_i$ in source network and $\mathcal{U}^t = \{u_j^t\}_j$ in target network. Each u_i^s has a ranking list or preference list $P(u_i^s)$ over all the user accounts in target network ($u_j^t \in \mathcal{U}^t$) based upon the input scores of different pairs. For example, in Figure 2(a), the preference list of node u_1^s is $P(u_1^s) = (u_1^t > u_2^t)$, indicating that node u_1^t is preferred by u_1^s over u_2^t. The preference list of node u_2^s is also $P(u_2^s) = (u_1^t > u_2^t)$. Similarly, we also build a preference list for each user account in the target network. In Figure 2(a), $P(u_1^t) = P(u_2^t) = (u_1^s > u_2^s)$.

The proposed Mna method for anchor link prediction is shown in Algorithm 1. In each iteration, we first randomly

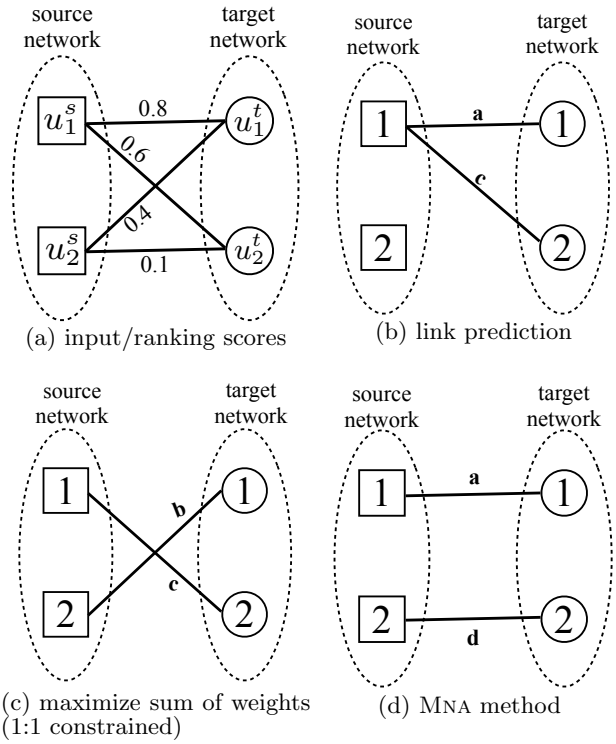


Figure 2: An example of anchor link inference by different methods. (a) is the input, ranking scores. (b)-(d) are the results of different methods for anchor link inference.

select a free user account u_i^s from the source network. Then we get the most prefered user node u_j^t by u_i^s in its preference list $P(u_i^s)$. We then remove u_j^t from the preference list, *i.e.*, $P(u_i^s) = P(u_i^s) - u_j^t$.

If u_j^t is also a free account, we add the pair of accounts (u_i^s, u_j^t) into the current solution set $\mathcal{A}'$. Otherwise, u_j^t is already occupied with u_p^s in $\mathcal{A}'$. We then examine the preference of u_j^t. If u_j^t also prefers u_i^s over u_p^s, it means that the pair (u_i^s, u_j^t) is a blocking pair. We remove the blocking pair by replacing the pair (u_p^s, u_j^t) in the solution set $\mathcal{A}'$ with the pair (u_i^s, u_j^t). Otherwise, if u_j^t prefers u_p^s over u_i^s, we start the next iteration to reach out the next free node in the source network. The algorithm stops when all the users in the source network are occupied, or all the preference lists of free accounts in the source network are empty.

4. EXPERIMENTS

4.1 Data Preparation

In order to evaluate the performance of the proposed approach for anchor link prediction, we tested our algorithm on two real-world social networks as summarized in Table 2. We chose Twitter and Foursquare as our data sources because public tweets and Foursquare tips can be easily collected by their APIs.

1) **Foursquare**: the first network we crawled is the Foursquare website, a representative location-based social network (LBSN). We collected a dataset consisting of 500 users

using breadth first search over the social graph and 7,504 tips of these users. For each tip, the location data (latitude and longitude) as well as the timestamp are available. Moreover, Foursquare network also provides data about whether one user is following or a friend of another user. These links can indicate the social relationship among the users.

2) **Twitter**: The second network we crawled is Twitter, an online social microblogging network. We collected 500 users which correspond to the 500 users in Foursqure and 741,529 tweets of the users. In Twitter network, all tweets include time data, and some tweets include location data. In total, we have 34,413 tweets with location data (latitude and longitude), which is about 4.6% of all the tweets we collected.

In order to conduct experiments, we pre-process these raw data to obtain the ground-truth of users' anchor links. In Foursquare network, we can collect some users' Twitter IDs in their account pages. We use these information to build the ground-truth of anchor links between user accounts across the two networks. If a Foursquare user has shown his/her Twitter ID in the website, we treat it as an anchor link between this user's Foursquare account and Twitter account.

Table 2: Properties of the Heterogeneous Social Networks

		network	
	property	**Twitter**	**Foursquare**
# node	user	500	500
	tweet/tip	741,529	7,504
	location	34,413	6,300
# link	friend/follow	5,341	2,934
	write	741,529	7,504
	locate	40,203	7,504

4.2 Comparative Methods

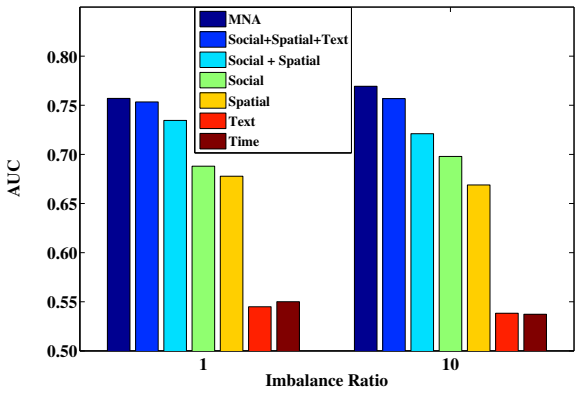
In order to study the effectiveness of the proposed approach, we compare our method with eight baseline methods, which are commonly used baselines including both supervised and unsupervised link prediction approaches. The compared methods are summarized as follows:

- *Multi-Network Anchoring(*MNA*)*: the proposed method in this paper. MNA can explicitly exploit four types of information from both networks to infer anchor links, *i.e.*, social, spatial, temporal and text data. In addition, MNA incorporates the one-to-one constraint in the inference process. We argue that by combining the four types of heterogeneous information as well as the one-to-one constraints, the performance of anchor link prediction can be effectively improved.

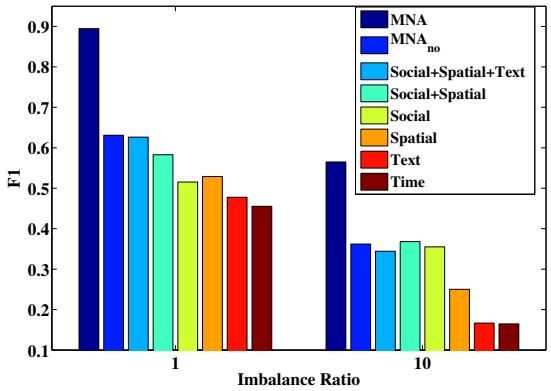
- MNA *without one-to-one constraint (*MNA_*no)*: our proposed method without the one-to-one constraint in the inference step. The label predictions of the base learners are directly used as final predictions for anchor link prediction.

- *Supervised link prediction methods*: in order to verify the effectiveness of different kinds of feature sets, we test supervised link prediction methods using four types of feature sets separately. 'Social' indicates the supervised link prediction method using social features only. 'Spatial' uses only spatial features. 'Time' uses temporal features. 'Text' uses text content features only. In order to verify the contribution of different features, we have also compared with different combinations of the heterogeneous feature sets as baseline methods. Details are shown in Figure 3.

- *Unsupervised Link Prediction Methods*: we also compare with a set of unsupervised link prediction methods: *Common Neighbor* (CN), *Jaccard Coefficient* (JC) and *Adamic/Adar* (AA). Since the original algorithms are designed for one single network. We modified these methods by treating any pair of anchor-linked accounts as one single node in the network and combining the social links in both networks into one single network among the users. Thus, we can use all the unsupervised methods to make predictions on each pair of user accounts.



(a) AUC



(b) F1

Figure 3: Performance of inferring anchor links with different sets of features.

For fair comparisons, LibSVM [6] of linear kernel with the default parameter is used as the base classifier for all the compared methods.

Evaluation Measures In order to evaluate the performance of anchor links prediction, we evaluate different ap-

Table 3: Performance comparison of different methods for inferring anchor links. We use different imbalance ratios in both training and test sets. (imbalance ration = # positive account pairs / # negative account pairs)

measure		imbalance ratio							
	methods	1	2	3	4	10	20	30	40
F1	Mna	**0.895±0.008**	**0.839±0.015**	**0.751±0.014**	**0.713±0.024**	**0.565±0.018**	**0.432±0.025**	**0.401±0.036**	**0.381±0.024**
	Mna_no	0.631±0.014	0.584±0.006	0.525±0.009	0.492±0.015	0.362±0.030	0.229±0.023	0.210±0.024	0.206±0.014
	Social	0.515±0.026	0.485±0.015	0.474±0.016	0.442±0.009	0.355±0.020	0.247±0.019	0.203±0.030	0.179±0.010
	Spatial	0.529±0.179	0.492±0.100	0.394±0.086	0.343±0.045	0.250±0.034	0.161±0.071	0.260±0.012	0.184±0.010
	Text	0.478±0.050	0.385±0.013	0.337±0.018	0.292±0.007	0.167±0.002	0.098±0.002	0.078±0.004	0.050±0.017
	Time	0.455±0.045	0.380±0.011	0.353±0.028	0.303±0.022	0.165±0.005	0.096±0.001	0.055±0.018	0.051±0.006
Prec.	Mna	**0.920±0.007**	**0.870±0.015**	**0.785±0.015**	**0.743±0.022**	**0.582±0.017**	**0.438±0.025**	**0.406±0.037**	**0.384±0.024**
	Mna_no	0.777±0.028	0.639±0.032	0.511±0.015	0.445±0.018	0.275±0.039	0.146±0.020	0.135±0.020	0.135±0.011
	Social	0.829±0.030	0.697±0.051	0.617±0.057	0.516±0.036	0.333±0.047	0.182±0.026	0.141±0.031	0.121±0.009
	Spatial	0.756±0.185	0.528±0.237	0.599±0.318	0.544±0.363	0.463±0.406	0.240±0.343	0.088±0.022	0.092±0.021
	Text	0.545±0.013	0.377±0.010	0.278±0.008	0.229±0.007	0.107±0.003	0.057±0.001	0.049±0.004	0.063±0.019
	Time	0.549±0.009	0.371±0.009	0.266±0.008	0.222±0.007	0.104±0.003	0.054±0.001	0.046±0.013	0.044±0.008
Rec.	Mna	**0.870±0.008**	**0.810±0.016**	**0.721±0.014**	**0.684±0.025**	**0.549±0.018**	0.425±0.025	0.396±0.036	0.377±0.024
	Mna_no	0.533±0.031	0.541±0.027	0.542±0.026	0.550±0.029	0.541±0.023	0.545±0.017	**0.485±0.029**	**0.435±0.014**
	Social	0.375±0.026	0.374±0.026	0.388±0.023	0.389±0.022	0.388±0.023	0.394±0.024	0.375±0.021	0.343±0.007
	Spatial	0.533±0.287	0.678±0.255	0.508±0.278	0.560±0.320	0.523±0.292	**0.659±0.244**	0.153±0.009	0.102±0.006
	Text	0.435±0.098	0.395±0.035	0.437±0.068	0.404±0.014	0.375±0.021	0.372±0.026	0.200±0.038	0.059±0.032
	Time	0.394±0.066	0.391±0.022	0.543±0.113	0.501±0.108	0.402±0.054	0.392±0.029	0.142±0.090	0.075±0.034
Acc.	Mna	**0.898±0.007**	**0.896±0.010**	**0.881±0.007**	**0.890±0.009**	**0.923±0.003**	**0.947±0.002**	0.962±0.002	0.970±0.001
	Mna_no	0.689±0.006	0.744±0.009	0.755±0.007	0.773±0.011	0.824±0.031	0.823±0.027	0.879±0.027	0.918±0.006
	Social	0.648±0.014	0.735±0.011	0.785±0.013	0.804±0.011	0.870±0.019	0.884±0.018	0.902±0.022	0.923±0.004
	Spatial	0.615±0.021	0.582±0.077	0.662±0.106	0.612±0.181	0.679±0.208	0.575±0.202	**0.972±0.000**	**0.978±0.000**
	Text	0.534±0.008	0.580±0.016	0.575±0.036	0.608±0.013	0.658±0.024	0.676±0.021	0.847±0.037	0.950±0.016
	Time	0.535±0.005	0.576±0.010	0.510±0.057	0.546±0.062	0.629±0.046	0.646±0.024	0.864±0.066	0.934±0.025
Auc	Mna	**0.757±0.010**	**0.771±0.008**	**0.751±0.011**	**0.752±0.009**	**0.769±0.012**	**0.758±0.009**	**0.762±0.014**	**0.775±0.010**
	Social	0.688±0.061	0.680±0.046	0.711±0.025	0.694±0.032	0.698±0.032	0.712±0.026	0.715±0.007	0.688±0.029
	Spatial	0.678±0.012	0.659±0.011	0.666±0.002	0.659±0.007	0.669±0.006	0.671±0.004	0.670±0.006	0.672±0.007
	Text	0.545±0.012	0.546±0.005	0.542±0.004	0.543±0.006	0.538±0.003	0.544±0.006	0.544±0.004	0.552±0.006
	Time	0.550±0.006	0.542±0.008	0.530±0.012	0.538±0.008	0.537±0.006	0.536±0.005	0.534±0.003	0.536±0.006
	CN	0.656±0.014	0.638±0.008	0.634±0.009	0.638±0.011	0.634±0.004	0.646±0.012	0.646±0.005	0.644±0.010
	JC	0.665±0.007	0.661±0.004	0.651±0.008	0.672±0.009	0.653±0.006	0.652±0.005	0.658±0.007	0.662±0.006
	AA	0.641±0.004	0.649±0.004	0.654±0.007	0.651±0.005	0.640±0.005	0.643±0.004	0.651±0.006	0.652±0.002

proaches in terms of F1-measure (F1), Precision (Prec.), Recall (Rec.), Accuracy (Acc.) and AUROC (Auc). The first 4 measures can evaluate the link prediction performances, while the AUROC evaluates the ranking performances. Since unsupervised link prediction methods (*i.e.*, CN, JC, AA) only predict a real-valued score without a label prediction for each pair of nodes , we only show the AUROC performances of unsupervised methods. Moreover, the only difference between Mna and Mna_no is on the constraints of label prediction, but they share the same ranking scores, *i.e.*, the real-value output of SVM. So for AUROC measure, we use Mna to represent both methods.

4.3 Performance of Anchor Link Prediction

In our experiments, we partition the users into two groups using 5-fold cross validation: one fold is used as training data, the remaining folds are used as testing data. We report the average results and standard deviations of 5-fold cross validation on the dataset.

In real-world networks, there are only a small number of known/labeled anchor links. In the first group of experiment, we study the performance of the proposed Mna method on anchor link prediction with different number of labeled anchor links. In each round of the cross validation, we randomly sample 10, 20, $\cdots$, 80 users from the training fold, and use them as the labeled anchor links. The results of all compared methods are reported in Table 4. The best performances on each of the evaluation criteria are listed in bold. It shows that when there are a small number of anchor links known in the two network, the proposed Multi-Network Anchoring (Mna) method consistently outperforms other baseline methods. This result supports the intuition of this paper: Multiple heterogeneous social networks can provide different types of information about the users. The anchor link prediction can be greatly improved by exploiting all four types of different information simultaneously.

In real-world link prediction problems, the data samples are usually imbalanced. In the second experiment setting, we test the performance of our method with imbalanced datasets. In each round of the cross validation, we sample pairs of user accounts as the data samples according to different imbalance ratios, *i.e.*, $\frac{\#\text{negative pairs}}{\#\text{positive pairs}}$. Table 3 shows the performances of each of the models under different imbalance ratios.

Moreover, in order to test the contribution of different type of features, we also tested the performances of baselines with different feature combinations. The result is shown in Figure 3. In Figure 3(a), we can see that when more types of features are used in the model, the better the performances we can get for anchor link prediction. In Figure 3(b), we notice that the performance of Mna is much better than Mna_no. It shows that by incorporating the one-to-one constraint in the inference process can further improve the performance of anchor link prediction.

4.4 Case Study

We show a case study to demonstrate the effectiveness of the proposed method by combining four types of heterogeneous information from two networks. In Figure 4, we show a case of five real-world users who have both Twitter and Foursquare accounts. These five users are socially connected in both networks, as shown in Figure 4(a). By considering this social information, we can significantly shrink the search space for anchor links if one or some of these users' accounts in both networks have already been labeled by anchor links. In Figure 4(b), we show the spatial distribution of different

Table 4: Performance comparison of different methods for inferring anchor links. We use different number of labeled anchor links in the training set.

measure	methods	number of labeled anchor links							
		10	20	30	40	50	60	70	80
F1	MNA	**0.735±0.055**	**0.828±0.035**	**0.843±0.036**	**0.849±0.027**	**0.862±0.012**	**0.881±0.008**	**0.881±0.011**	**0.896±0.008**
	MNA_no	0.502±0.083	0.510±0.095	0.522±0.032	0.584±0.021	0.584±0.042	0.583±0.030	0.616±0.027	0.609±0.016
	Social	0.031±0.063	0.190±0.110	0.334±0.044	0.382±0.030	0.396±0.026	0.445±0.023	0.447±0.013	0.501±0.019
	Spatial	0.259±0.317	0.430±0.197	0.455±0.267	0.425±0.203	0.592±0.161	0.593±0.160	0.597±0.157	0.680±0.004
	Text	0.466±0.018	0.493±0.038	0.457±0.057	0.490±0.057	0.435±0.018	0.437±0.022	0.460±0.016	0.438±0.009
	Time	0.559±0.011	0.553±0.021	0.529±0.036	0.485±0.080	0.523±0.061	0.492±0.069	0.507±0.051	0.455±0.063
Prec.	MNA	**0.785±0.052**	**0.866±0.030**	**0.877±0.031**	**0.884±0.023**	**0.894±0.010**	**0.909±0.006**	**0.910±0.008**	**0.921±0.006**
	MNA_no	0.559±0.034	0.654±0.080	0.680±0.069	0.670±0.019	0.717±0.054	0.727±0.033	0.715±0.034	0.754±0.032
	Social	0.173±0.346	0.647±0.354	0.798±0.076	0.855±0.037	0.822±0.036	0.837±0.048	0.821±0.029	0.828±0.039
	Spatial	0.223±0.274	0.818±0.218	0.544±0.316	0.826±0.205	0.642±0.172	0.660±0.157	0.678±0.159	0.595±0.021
	Text	0.530±0.004	0.543±0.031	0.530±0.026	0.523±0.011	0.554±0.016	0.544±0.020	0.556±0.012	0.539±0.004
	Time	0.530±0.007	0.525±0.006	0.527±0.016	0.521±0.012	0.530±0.013	0.526±0.010	0.529±0.005	0.529±0.020
Rec.	MNA	**0.692±0.057**	**0.794±0.039**	**0.811±0.040**	**0.816±0.030**	**0.832±0.013**	**0.854±0.009**	**0.854±0.013**	**0.871±0.010**
	MNA_no	0.482±0.143	0.460±0.173	0.429±0.049	0.520±0.037	0.508±0.098	0.491±0.052	0.547±0.055	0.513±0.035
	Social	0.017±0.035	0.119±0.081	0.215±0.045	0.247±0.028	0.262±0.023	0.304±0.023	0.307±0.012	0.360±0.025
	Spatial	0.316±0.395	0.437±0.345	0.504±0.352	0.417±0.331	0.711±0.278	0.674±0.263	0.655±0.249	0.797±0.040
	Text	0.417±0.028	0.467±0.109	0.420±0.120	0.479±0.133	0.360±0.028	0.368±0.037	0.393±0.026	0.370±0.015
	Time	0.593±0.027	0.587±0.049	0.539±0.079	0.478±0.162	0.533±0.114	0.474±0.115	0.495±0.089	0.410±0.108
Acc.	MNA	**0.752±0.050**	**0.836±0.032**	**0.849±0.033**	**0.855±0.025**	**0.866±0.011**	**0.885±0.008**	**0.884±0.010**	**0.898±0.007**
	MNA_no	0.544±0.021	0.589±0.020	0.609±0.026	0.631±0.011	0.646±0.010	0.651±0.010	0.662±0.006	0.671±0.008
	Social	0.507±0.015	0.533±0.022	0.576±0.004	0.602±0.008	0.602±0.009	0.622±0.011	0.620±0.008	0.642±0.007
	Spatial	0.530±0.039	0.578±0.010	0.576±0.045	0.586±0.019	0.584±0.010	0.604±0.017	0.618±0.020	0.625±0.016
	Text	0.524±0.004	0.530±0.027	0.517±0.019	0.518±0.007	0.534±0.008	0.528±0.010	0.539±0.008	0.527±0.002
	Time	0.533±0.006	0.528±0.006	0.525±0.011	0.518±0.011	0.528±0.007	0.525±0.012	0.526±0.004	0.523±0.015
AUC	MNA	**0.556±0.029**	**0.640±0.040**	**0.657±0.021**	**0.688±0.021**	**0.705±0.008**	**0.709±0.012**	**0.721±0.008**	**0.735±0.013**
	Social	0.507±0.015	0.534±0.021	0.572±0.029	0.628±0.029	0.627±0.039	0.651±0.021	0.670±0.029	0.667±0.024
	Spatial	0.549±0.061	0.621±0.046	0.602±0.043	0.658±0.005	0.651±0.017	0.660±0.006	0.670±0.008	0.671±0.008
	Text	0.529±0.005	0.533±0.031	0.510±0.043	0.530±0.003	0.544±0.006	0.537±0.009	0.540±0.011	0.543±0.003
	Time	0.538±0.006	0.539±0.011	0.534±0.017	0.519±0.024	0.543±0.006	0.531±0.015	0.531±0.006	0.531±0.021
	CN	0.527±0.005	0.541±0.004	0.581±0.007	0.591±0.003	0.599±0.004	0.617±0.012	0.634±0.005	0.627±0.006
	JC	0.528±0.007	0.546±0.004	0.577±0.010	0.593±0.007	0.608±0.010	0.616±0.012	0.630±0.009	0.631±0.004
	AA	0.524±0.004	0.552±0.008	0.575±0.007	0.585±0.012	0.601±0.010	0.610±0.009	0.619±0.007	0.631±0.009

users on both networks. We can see that the spatial distributions of the same user are pretty similar to each other. Michelle is mainly located in the middle states of America, when sending tweets and foursquare tips. The spatial distributions of her foursquare account and twitter accounts are pretty similar. In Figure 4(c), we show the temporal distribution of the users. We can see that Tristan's temporal activities across both Twitter account and Foursquare account are very consistent, and his distribution is very different from Lisa's temporal activity patterns. In Figure 4(d), we show some frequently used words by the users, where the choices of words of the same user can be pretty consistent. For example, Andrew seems to prefer to use 'awsm' instead of 'awesome' when writing tweets and tips.

5. RELATED WORK

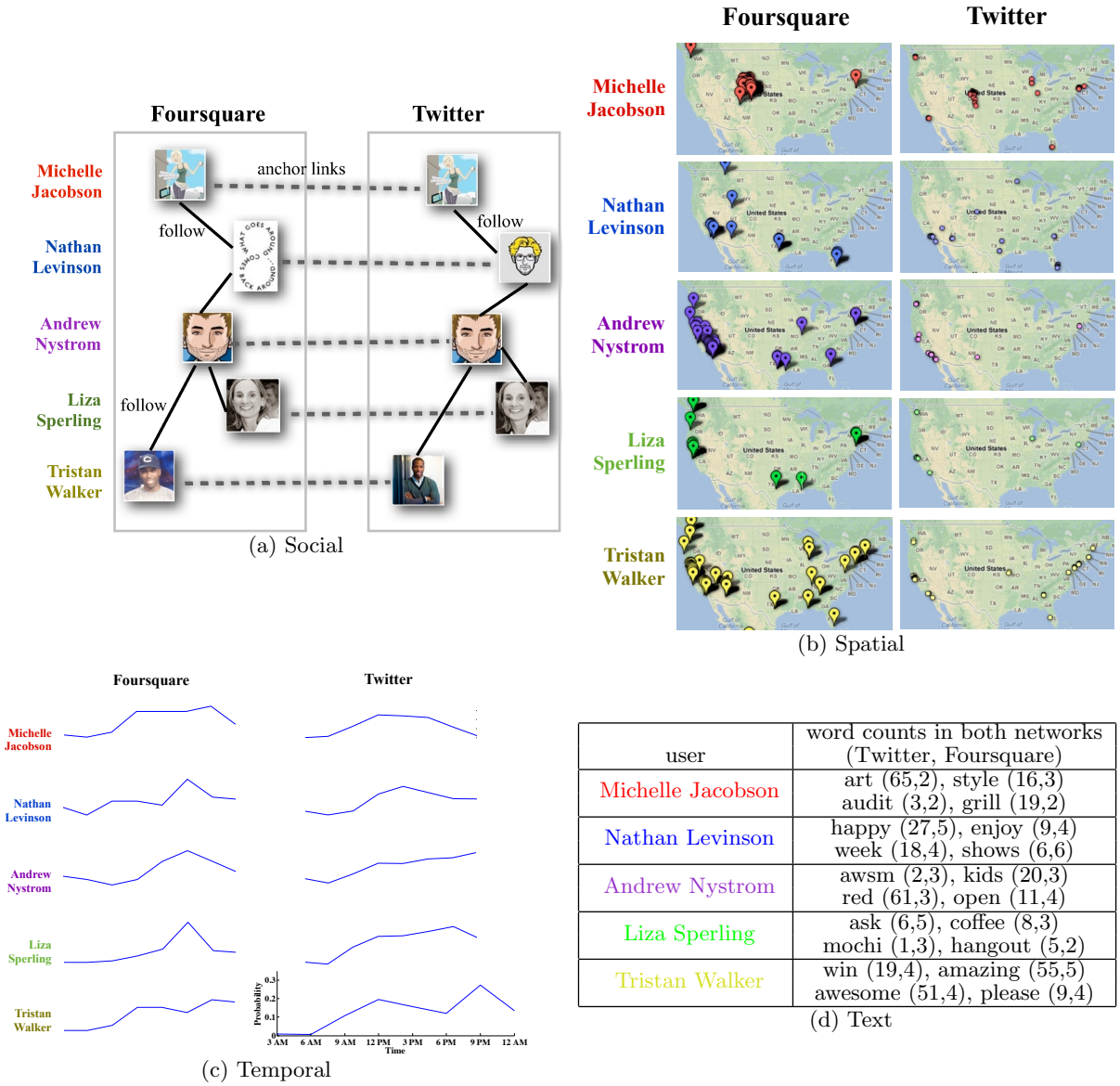
Social network analysis [16, 12], especially the link prediction problem in social networks, has been intensively studied in recent years [13, 10, 23]. Typically some similarity measures between pair of nodes are used. Upon whether considering the label information, there are two types of approaches: unsupervised and supervised. Liben-Nowell and Kleinberg [13] developed unsupervised link prediction methods based upon several topological features of a co-author network. Many supervised link prediction methods have also been proposed in recent years, [10], where the features used in unsupervised approaches can be directly used to train a binary classification model for link prediction. There are many other recent efforts on link prediction problem in social networks. Lichtenwalter *et. al.* [14] have a detailed discussion over different challenges of link prediction problem. Scellato *et. al.* [18] proposed to use place features for link prediction in location-based social networks. [2] proposed a supervised random walk method for link predictions in social networks. In addition, another line of research works study the link prediction problems on multiple networks or domains [5, 7, 20, 24, 21].

Network alignment problem has also been studied by many works in recent years, which has many applications bioinformatics [11, 19]. Bayati *et. al.* [3] proposed to use belief propagation to solve sparse network alignment problems. Most network alignment approaches focus on finding approximate isomorphisms between two graphs under unsupervised settings. Because the intractability of the problem, existing methods usually rely on practical heuristics to solve the alignment problem.

Our work is also related to other lines of research. Location-based social networks have been researched recent years [18, 25], which mainly focus on single network setting. Previous works have also explored the multi-network problems, such as user identification [22], profile matching [17] and matching user footprints[15]. These research works focus mainly on matching social network users based upon user profile information, such as sharing similar user names, sharing email address, *etc.*. Our approach assumes heterogeneous information in the networks is available, and focus on using social links, location distribtutions and temporal distributions to infer the account similarity. User profiles (*e.g.* username, email) are excluded in our study.

6. CONCLUSION

In this paper we have described and studied the problem of inferring anchor links across multiple heterogeneous social networks. We have studied two real-world social networks, Foursquare and Twitter, finding the correspondence of different users accounts. Different from previous works in link prediction and network alignment, we assumed that the anchor links is an one-to-one relationships between the user



user	word counts in both networks (Twitter, Foursquare)
Michelle Jacobson	art (65,2), style (16,3) audit (3,2), grill (19,2)
Nathan Levinson	happy (27,5), enjoy (9,4) week (18,4), shows (6,6)
Andrew Nystrom	awsm (2,3), kids (20,3) red (61,3), open (11,4)
Liza Sperling	ask (6,5), coffee (8,3) mochi (1,3), hangout (5,2)
Tristan Walker	win (19,4), amazing (55,5) awesome (51,4), please (9,4)

(d) Text

Figure 4: Case study: five real-world users with their social, spatial, temporal and text distributions.

accounts in two networks, and we know some existing anchor links before the inference. By explicitly consider the users heterogeneous data within the networks, *i.e.*, social, spatial, temporal and text information, our method can effectively predict the anchor links w.r.t. one-to-one constraint across multiple heterogeneous social networks.

7. ACKNOWLEDGEMENTS

This work is supported in part by NSF through grants IIS-0905215, CNS-1115234, IIS-0914934, DBI-0960443, and OISE-1129076, and US Department of Army through grant W911NF-12-1-0066.

8. REFERENCES

[1] L. Adamic and E. Adar. Friends and neighbors on the web. *Social Networks*, 25, 2003.

[2] L. Backstrom and J. Leskovec. Supervised random walks: predicting and recommending links in social networks. In *Proceedings of the 5th International Conference on Web Search and Web Data Mining*, pages 635–644, Hong Kong, China, 2011.

[3] M. Bayati, M. Gerritsen, D. Gleich, A. Saberi, and Y. Wang. Algorithms for large, sparse network alignment problems. In *Proceedings of The 14th IEEE International Conference on Data Mining*, pages 705–710, Miami,FL, 2009.

[4] I. Bhattacharya and L. Getoor. Collective entity resolution in relational data. *ACM Transactions on Knowledge Discovery from Data*, 1(1), 2007.

[5] B. Cao, N. Liu, and Q. Yang. Transfer learning for collective link prediction in multiple heterogeneous domains. In *Proceedings of the 27th International Conference on Machine Learning*, pages 159–166, Haifa, Israel, 2010.

[6] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[7] Y. Dong, J. Tang, S. Wu, J. Tian, N. Chawla, J. Rao, and H. Cao. Link prediction and recommendation across heterogeneous social networks. pages 181–190, Brussels, Belgium, 2012.

[8] L. Dubins and D. Freedman. Machiavelli and the gale-shapley algorithm. *The American Mathematical Monthly*, 1981.

[9] L. Getoor and C. Diehl. Link mining: a survey. *SIGKDD Explorations*, 7, 2005.

[10] M. Hasan, V. Chaoji, S. Salem, and M. Zaki. Link prediction using supervised learning. In *SDM workshop on Link Analysis, Counterterrorism and Security*, 2006.

[11] G. Klau. A new graph-based method for pairwise global network alignment. *BMC Bioinformatics*, 10, 2009.

[12] J. Leskovec, L. Backstrom, R. Kumar, and A. Tomkins. Microscopic evolution of social networks. In *Proceedings of The 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 462–470, Las Vegas, NV, 2008.

[13] D. Liben-Nowell and J. Kleinberg. The link prediction problem for social networks. In *Proceedings of the 2003 ACM CIKM International Conference on Information and Knowledge Managemen*, pages 556–559, New Orleans, LA, 2003.

[14] R. Lichtenwlter, J. Lussier, and N. Chawla. New perspectives and methods in link prediction. In *Proceedings of The 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 243–252, Washington, DC, 2010.

[15] A. Malhotra, L. Totti, W. Meira, P. Kumaraguru, and V. Almeida. Studying user footprints in different online social networks. In *CoRR*, 2013.

[16] M. Newman. Clustering and preferential attachments in growing networks. *Physical Review Letters*, 2001.

[17] E. Raad, F. Dijon, R. Chbeir, and A. Dipanda. User profile matching in social networks. In *Proceedings of The 13th International Conference on Network-Based Information Systems*, pages 297–304, Takayama, Japan, 2010.

[18] S. Scellato, A. Noulas, and C. Mascolo. Exploiting place features in link prediction on location-based social networks. In *Proceedings of The 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1046–1054, San Diego, CA, 2011.

[19] R. Singh, J. Xu, and B. Berger. Pairwise global alignment of protein interaction networks by matching neighborhood topology. In *Proceedings of the 11th Annual International Conference on Research in Computational Molecular Biology*, San Diego, CA, 2007.

[20] J. Tang, T. Lou, and J. Kleinberg. Inferring social ties across heterogeneous networks. In *WSDM*, 2012.

[21] J. Tang, S. Wu, J. Sun, and H. Su. Cross-domain collaboration recommendation. In *Proceedings of The 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1285–1293, Beijing, China, 2012.

[22] J. Vosecky, D. Hong, and V.Y. Shen. User identification across multiple social networks. In *Proceedings of The 1st International Conference on Networked Digital Technologies*, pages 360–365, Ostrava, Czech Republic, 2009.

[23] C. Wang, V. Satuluri, and S. Parthasarathy. Local probabilistic models for link prediction. In *Proceedings of The 12th IEEE International Conference on Data Mining*, pages 322–331, Omaha, NE, 2007.

[24] Y. Yang, N. Chawla, Y. Sun, and J. Han. Link prediction in heterogeneous networks: influence and time matters. In *Proceedings of The 12th IEEE International Conference on Data Mining*, Brussels, Belgium, 2012.

[25] M. Ye, D. Shou, W. Lee, P. Yin, and K. Janowicz. On the semantic annotation of places in location-based social networks. In *Proceedings of The 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 520–528, San Diego, CA, 2011.

[26] B. Zadrozny and C. Elkan. Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of The 2nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 694–699, Edmonton, AB, 2002.