Spatio-Temporal Prediction of Social Connections

Guolei Yang
Iowa State University
Ames, Iowa, USA 50011
yanggl@iastate.edu

Andreas Züfle
George Mason University
Fairfax, Virginia, USA 22030
azufle@gmu.edu

ABSTRACT

It is long known that a user's mobility pattern can be affected by his social connections. Users tend to visit same locations visited by their friends. In this paper we investigate the inverse problem: How does a set of users' trajectory reflects their social connections. To this end, we define the social connection prediction problem. Given two users, predict the probability that they are friends by mining their historical trajectories. A naive method to do so is to exam how often the two users visits the same location at the same time, which suffers from the problem that different locations/times may have different predictive power. We propose a comprehensive prediction model that is able to capture this difference between locations and time slots. To demonstrate its effectiveness, we trained the proposed model using the publicly available Foursquare dataset. The result shows the proposed model is able to predict existence of social connections between randomly selected users significantly more accurate comparing with the naive method.

CCS CONCEPTS

•Information systems → Location based services; Data mining; •Computing methodologies → *Machine learning;*

KEYWORDS

Location-Based Social Network, social connection prediction, feature selection, spatio-temporal data

ACM Reference format:
Guolei Yang and Andreas Züfle. 2016. Spatio-Temporal Prediction of Social Connections. In *Proceedings of ACM Conference, Washington, DC, USA, July 2017 (Conference'17),* 5 pages.
DOI: 10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

In the past decade, with the rise of Location-Based Social Networks (LBSN), huge amount of geo-spatial data is collected on a daily basis. For example, the Foursquare[14] dataset contains more than 30 millions of self-reported check-ins from thousands of user around the world. As a result, it becomes possible to mine spatio-temporal data and study human mobility pattern at unprecedented large scale.

Conference'17, Washington, DC, USA

DOI: 10.1145/nnnnnnn.nnnnnnn

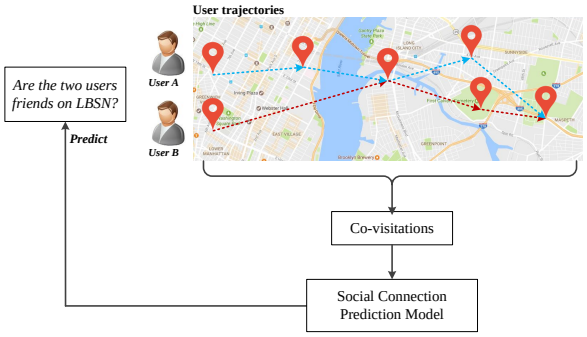

Figure 1: An illustration of spatio-temporal social connection prediction.

Several studies on human mobility pattern reveal that a user's movement can be affected by his social connections [2, 15]. For example, a group of close friends tend to check-in to the same locations at the same time period. As such, it is possible to predict a user's future movement by mining the historical trajectory of his friends on LBSN. These studies has since inspired a seriers of research efforts towards the prediction of future individual movements (e.g., [4, 6, 9, 11]). Another research direction (e.g., [13]) focuses on exploring historical trajectories to identify similar users, which can be used for friend recommendation and community detection on LBSN.

Towards the goal of a more thorough understanding of human mobility patterns, we propose to investigate the inverse problem: How does a set of users' trajectory reflects their social connections. In particular, we define the social connection prediction problem: Given the trajectories of two LBSN users u_i and u_j, we aim to model the probability that u_i and u_j are friends on the LBSN using their trajectories. Social connection prediction is a long standing research topic. Most existing methods relies on link prediction techniques, which exploit a user's profile and existing social connections to make predictions of hiding links between users, but not users' trajectory. The focus of this paper is not to compete with, but to supplement existing methods by exploring a new dimension of data source.

A straightforward way to predict the social connection, or the lack thereof, between two users is to exam the *spatio-temporal overlap* of their trajectories, i.e., find events where the two users visit the same location at the same time on their trajectories. We define such an event as a *co-visitation* of the two users. The assumption is, if two users frequently visits the same location during the same time peroid, they might be friend with each other. Thus the occurance of co-visitations could reflect when and where they were meeting. The same assumption is used to identify similar users

in [13]. Algorithms such as co-location mining [12] can be used to discover co-visitations among users.

Although the above assumption is reasonable, this naive solution suffers from two problems. First, it treats all locations equally in predicting social connections, which is not realistic. For example, if two users frequently meet at private locations like someone's house, or a small coffee shop, it is very likely that they know each other. However, if they both check-in to the same Walmart supermarket after work, it might be just an coincidence simply because there it is the only supermarket near their home. Second, this method ignores the time difference of check-ins behaviours. If two users both check-in to a restaurant at 6:00pm, it is not as significant as two users visit the same location at 10:00pm. This is because most customer of the restaurant may choose to dine there around 6:00pm, but if two users both decide to check-in there at 10:00pm, the chance that it is purely an coincidence is relatively lower. Although the technique proposed in [13] considered the impact of different granularity of locations (e.g., the same state v.s. the same city), it does not explicitly distinguish the predictive power of different locations/time for different users.

We propose to employ a more comprehensive methodology to study the social connection prediction problem. Unlike the naive solution, we assume different locations and different time slots have different predictive power. As such, we propose a social connection prediction model in which the predictive power of each location and time slot are treated as latent variables. The proposed model is based on a novel data structure termed *Spatio-Temporal Co-visitation Matrix*. Additionally, our model also take into consideration the geographic distance between the user's home/work location to the co-visitation locations. The latent variables in the model are then learnt with the Foursquare dataset. Using the users' social connections on Foursquare as ground truth, we show that the proposed model outperforms the naive algorithm that counts only the number of co-visitations. We summarize our contributions as follows:

- We study how the trajectories of a set of users reflect their social connections. To this end, we define the social connection prediction problem: Given the trajectories of two LBSN users u and v, we aim to model the probability that u and v are friends on the LBSN.
- Our key observation is: different locations and time may have different predictive power, which is in accordance with common sense. As such, we propose a social connection prediction model that is able to capture this difference among locations and times using latent variables.
- We demonstrate effectiveness of the proposed model using the Foursquare dataset. The result shows the proposed method outperforms the naive trajectory overlap based solution in prediction accuracy.

The rest of the paper is organized as follows: Related works are summarized in Section 2. We formally define our problem and give an overview of our methodology in Section 3. Section 4 present the key data structure and the proposed model. Experiment results are showed and analysed in Section 5. And finally, Section 6 concludes the paper.

2 RELATED WORK

The spatio-temporal social connection prediction problem we study in this paper is directly related to link prediction problem on social networks. Given the snapshot of a social network at time t, the goal of link prediction is to predict links, i.e., social connections, that will emerge at a later time, or to identify missing links at t. Such missing links could be the result of privacy settings, e.g., a user may want to hide his friend list from the general public.

Existing works in the field mainly explore two types of information in predicting links: 1) Network structure, i.e., existing social connections, and 2) node attributes such as user profiles. We briefly summarize some representative works. The relational learning [8, 10, 17] and matrix factorization-based [7] techniques both leverage attribute information for link prediction. The Supervised Random Walk (SRW) technique proposed in [1] combines networks structure and edge attributes to improve prediction accuracy, but does not fully explore node attributes. In [16], network structure and node attributes are integrated with a Social Attribute Network (SAN) model, which is later generalized in [5] to both predict links and infer missing attributes.

Our problem is also closely related to [13], which proposes to explore trajectory data to identify similar users. Their goal is to find users who share similar interests in locations, which serves as a friend recommendation tool on LBSN. The proposed technique employs clustering over users location history to identify similar users. In contrast, our work focus on studying the predictive power of trajectories in terms of reflecting existing or missing social connections among users. And our methodology is to model the probability of the existence of such social connections between two users. Form this perspective, our work intends to complement existing studies on human mobility patterns.

3 OVERVIEW

3.1 Problem Statement

We define a user's trajectory as a series of timestamped check-ins, where each check-in indicates the exact place (i.e., a restaurant, a coffee shop, etc.) the user visits, instead of a geo-graphical coordinate. The Foursquare dataset is an example of such trajectory that consists of self-reported check-ins. Note that coordinate-based trajectory can be converted into such check-ins by joining the coordinates with a database of Point-of-Interests (PoI), such as provided by Open-Stree Map. For simplicity, we consider only check-in-based trajectory in this paper. We formally define the notion of *Check-in* and *User Trajectory* as follows.

Definition 3.1 (Check-in). Let $\mathbb{U}$ denote a set of unique user identifiers, $\mathbb{L}$ denote a set of locations, and $\mathbb{T}$ denote the time domain. A check-in c is a triple $(u, l, t) \in \mathbb{U} \times \mathbb{L} \times \mathbb{T}$, which indicates the user u has visited l at time t.

Definition 3.2 (User-trajectory). Let $\mathbb{C}$ be a collection of check-ins and $u \in \mathbb{U}$ a user, then the set $C_u := \{(u', l, t) \in \mathbb{C} | u = u'\}$ is the user-trajectory (or simply trajectory) of u.

The proposed social connection prediction model is based on the concept of *Co-visitation*, which is defined as follows:

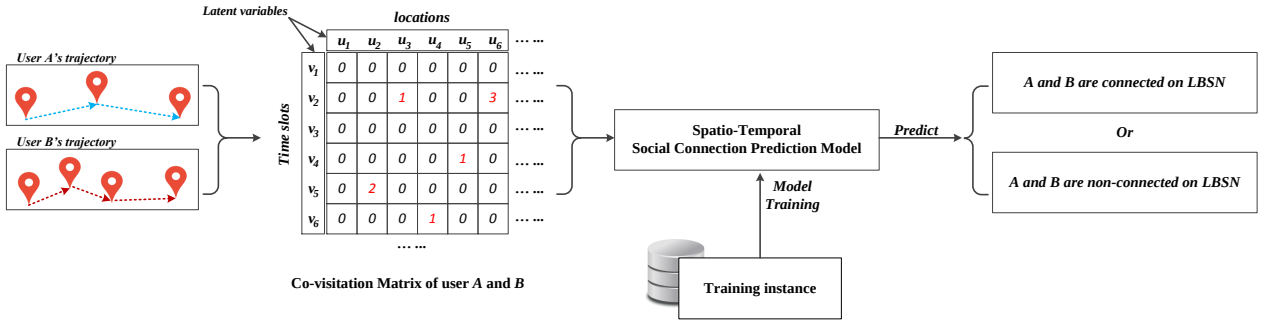

Figure 2: General steps of the proposed method.

Definition 3.3 (Co-visitation). A co-visitation of two users u_i and u_j to a location l is defined as the event that u_i and u_j report two check-ins (u_i, l, t_i) and (u_j, l, t_j) respectively, where $|t_i - t_j| \leq \tau$.

Here τ is an experience-based parameter called the co-visitation time window. We formulate the social connection prediction problem as a classification problem. Given the trajectory of two users u_i and u_j, the goal is to assign the pair of users (u_i, u_j) into one of the two classes: *Connected* or *Not-connected*.

Note that the above problem formulation implicitly assumes the social connection is mutual, i.e., if u_i is a friend of u_j, then u_j is also a friend of u_i. This type of social connection is common on most LBSN like Facebook. For such connections, the class of a pair of users is not order sensitive. However, on some LBSN the social connection can be one-way. For example, on Foursquare, a user can choose to "follow" other users and thus formulates a one-way social connection, where one user is the *follower* and the other being the *followee*. The one-way social connection can be model by treating (u_i, u_j) and (u_j, u_i) as different instance which can be assigned into different classes.

3.2 Methodology

We model the probability of the existence of social connection between two users based on the hypothesis that socially connected users tend to visit same locations at same time periods, which is defined as co-visitations. However, we observes that in reality not all co-visitations are equally important in terms of predicting user's social connections. We propose a three step model learning process to capture this difference (Figure 2).

- **Co-visitation Matrix formulation** Given the trajectories of two users u_i and u_j, we first convert their trajectories into a spatio-temporal co-visitation matrix that records the time and location of their co-visitations.
- **Probability estimation** The probability that u_i and u_j are socially connected is computed based on their co-visitation matrix.
- **Model learning** The latent variables in the model are estimated by optimization a loss function, which measures the prediction error between actual class and predicted class for each pair of users in the training set.

We present details of these three steps in the next section.

4 PROPOSED MODEL

4.1 Spatio-Temporal Co-visitation Matrix

Given the trajectories of two users C_{u_a} and C_{u_b}, their spatio-temporal co-visitation matrix $M(a, b)$ is a $m \times n$ matrix where m is the total number of location in $\mathbb{L}$ and n the number of time slots. The i-th row in $M(a, b)$ corresponds to the i-th location while the j-th column corresponds to the j-th time slot. As such, if the two users had $x \in \mathbb{N}$ co-visitation to the i-th location that occurred within the j-th time slot, $M(a, b)_{i,j}$ is set to x and otherwise 0. Figure illustrates a co-visitation matrix generated for two users. Note that the co-visitation graph is usually highly sparse.

The granularity of locations and time slots used to build the co-visitation matrix can be adjusted as needed, i.e., each location can be an exact PoI or a geographic region with arbitrary size. The time slot can be hours or days. The purpose of this mechanism is to provide the users with the flexibility to control the number of model parameters need to be learnt from training data. For larger dataset, more locations and time slots can be used. However, if the number of labelled instance is limited, using a large number of parameters may risk over-fitting the model. Without loss of generality, in this paper, we use the following granularity settings. 1) Each location is a specific PoI (i.e., a restaurant, a coffee shop, etc.), but the total number of PoIs used is limited to 100. 2) We partition each day into 5 time slots:(12:00am to 6:00am), (6:01am to 10:00am), (10:01am to 2:00pm), (2:01pm to 5:00pm), (5:01pm to 8:00pm), and (8:01pm to 11:59pm). Note that these time slot are not evenly partitioned. Instead, we choose this typical time slot that reflects different period of a day for work or social events. As such, we use a total of 35 time slots, because each day of a week has 5 slots.

Efficient co-visitation detection: The co-visitations of two given users can be detected by exhaustive searching, i.e., for each check-in of a user u_a, exams every check-in of u_b to see if they formulate a co-visitation. This is obviously inefficient especially when the number of user are large in generating the training instances. Here we design a more efficient co-visitation detection algorithm. We describe the algorithm step by step:

- **1)** Each check-in $c = (u, l, t)$ is converted into two tuples $c_s = (u, l, t - \frac{\tau}{2})$ and $c_e = (u, l, t + \frac{\tau}{2})$. As such, each check-in is extended to a time period $[t - \frac{\tau}{2}, t + \frac{\tau}{2}]$. Two check-ins to the

same location l is a co-visitation if and only if their time period overlaps.
- **2)** All tuples are sorted by the timestampe in ascending order.
- **3)** For each possible co-visitation location, initialize an empty list. Initialize the co-visitation matrix to be all zeros.
- **4)** The algorithm then performs a running count by scanning through the sorted tuples one by one.
- **4.i)** When c_s is encountered for some check-in c, it is is added to the list of location l. If l is not empty, a co-visitation is detected. Update the co-visitation matrix accordingly.
- **4.ii)** When c_e is encountered for some check-in c, remove c_s of the same check-in from the list of location l.

Note that the above algorithm has log-linear complexity to the total number of check-ins. More importantly, it is able to detect co-visitation for a set of users in a parallel manner. Thus it is much faster than exhaustive search, whose complexity is quartic to the number of check-ins for each pair of users.

4.2 Social Connection Prediction Model

In our model, both locations and time slots are mapped into a 1-dimensional latent space. We use $U \in \mathbb{R}^m$ and $V \in \mathbb{R}^m$ to denote the latent variables for locations and time slots, where $u_i \in U$ can be seen as a weight that measures the significance of the i-th locations. Similarly, $v_j \in V$ measures the significance of the j-th time slot. Given the co-visitation matrix $M(a, b)$, we can then estimate that how likely u_a and u_b are socially connected using a weighted sum over the matrix, defined as follows:

$$s(a, b) = \sum_{i=1}^{m} \sum_{j=1}^{n} u_i v_j M(a, b)_{i,j} \tag{1}$$

where $s(a, b)$ can be seen as a "score". The higher the score is, the more likely u_a and u_b are socially connected. We employ the sigmoid function to convert $c(a, b)$ into an estimated probability for the classification problem:

$$\hat{Pr}((u_a, u_b) \text{ is connected}) = \frac{1}{1 + e^{-c(a,b)}} \tag{2}$$

If the predicted probability is higher than a decision threshold, denoted by λ, the user pair is classified as *connected*, otherwise *non-connected*.

The above model, however, does not take into consideration the factor of geographic distance between locations. The same location may have different significance for users lives in different area. The geographic distance between a user's home/work and the co-visitation locations can been seen as a personalized parameter to adjust the significance of a location. To this end, we modify Equation 1 by adding the *distance coefficients* $W = \{w_1, w_2\}$ to our model:

$$s(a, b) = \sum_{i=1}^{m} \sum_{j=1}^{n} \Big(u_i v_j + w_1 D(i, a) + w_2 D(i, b) \Big) M(a, b)_{i,j} \tag{3}$$

Here, w_1 and w_2 are the two distance coefficients, and $D(i, a)$ measures the geographic distance between the i-th locations and u_a's home base. For simplicity, we use the geographic center of u_a's all check-ins as the estimated home base coordinate. Nevertheless, more complex method such as the one proposed in [2] can also be used for more strict estimation. Note that by introducing the distance coefficients we only added two more parameters into our model, but it allows the classification results to be "personalized" to some extend by involving the two user's home base locations into the model.

4.3 Model Learning

Parameters in Equation 3 can be learned from a set of labelled training data by optimization the solving the function:

$$\arg\min_{U,V,W} \sum_{\forall u_a, u_b \in \mathbb{U}} E(p_{a,b}, \hat{p}_{a,b}) + \Theta(U, V) \tag{4}$$

In the above function, $E()$ denote a loss function that measures the prediction error. In this paper we use the indicator function as loss function, which is commonly used for classification problems. $p_{i,j}$ is the label of a training instance (u_a, u_b) while $\hat{p_{a,b}}$ is the predicted result using the proposed model. Finally, $\Theta(U, V)$ is the regularization term, defined as:

$$\Theta(U, V) = \frac{\lambda_u}{2} \|U\|_2^2 + \frac{\lambda_v}{2} \|V\|_2^2 \tag{5}$$

The regularization term is in place to prevent the model from over-fitting. The regularization coefficients λ_u and λ_v are selected through a cross-validation process in our experiments.

Note that in the co-visitation matrix, we assigned a latent variable to each location and each time slot. As a result, it may appears a total number of $m + n + 2$ parameters need to be learnt from the training data. However, the actual number of parameters can be much smaller. This is because the the co-visitation matrix is usually highly sparse. If the two users have never reported a co-visitation to certain location l, then the corresponding latent variable does not need to be learnt. The value of the varible is simply set to 0. Similarly, the latent variable for a time slot is set to 0 if no co-visitations occurred within the time slot. Eventually, only those locations and time slots that are significant will have a non-zero latent variable value. This makes the model parameters easily interpretable by human.

5 EVALUATION

In this section, we report the preliminary experiment results of the proposed methodology on the trajectory of a selected subset of Foursquare users.

5.1 Dataset Description

We evaluate the proposed model on the widely-used Foursquare check-in dataset [14]. In our experiments, we mine the check-in data from two of the most popular cities, including New York City (NYC) and Tokyo. The dataset contains about 227,428 check-ins reported in NYC and 573,703 in Tokyo. The check-ins were collected for about 10 month. From each check-in, we extract the user ID, location ID, and a timestamp. Using the user ID or location ID, we retrieve the profile of that user or location on Foursquare. The user profile include the social connection between users ("follower - followee") and the location profile includes its category (*Food, Coffee, Nightlife, Fun, and Shopping*), coordinates, and user rating. The check-ins are grouped by user ID/location ID and sorted by their timestamps.

For our experiments, we select a subset of users that satisfy the following conditions:

- **Check-in Active** Actively report check-ins on a daily basis.
- **Socially Active** The user have followers and/or also follows others.

These two conditions are in place to filter out the users who do not have enough data or lack the ground truth to test the proposed model. To ease the user selection process, we applied community detection [3] and selected two communities, each contains approximately 150 users who satisfy both conditions, from the two cities. Each user have on average 19 social connections. We show in the following subsections that this small set of users is sufficient to demonstrate the effectiveness of the proposed model.

5.2 Experiment Design

For comparison purpose, we have implemented the following schemes:

- **Random** This scheme randomly assigns a user pair as friends or non-friends, each with a probability of 50%.
- **Naive** This scheme simple counts the number of co-visitations of two users. If the number is higher than a threshold, the two users are predicted to be friends, and otherwise non-friends. The threshold is set to be the average number of co-visitations of each pair of friends among the selected users.
- **Proposed** The proposed co-visitation matrix-based model.

5.3 Results

6 CONCLUSION

In this paper, we study the predictive power of user trajectories in reflecting their social connections. Based on the hypothesis that friends tend to visit same locations at same time, we propose to model the probability of that social connection exists between two users using their co-visitations. We propose a three step model learning process. First, a co-visitation feature vector is generated based on the trajectory of two users. We then employ feature selection to filter out less significant co-visitations from the feature vector. Finally, we explore several statistic models in order to find the one that is able to yield the best performance for our problem. In our preliminary experiments using a subset of users selected from the Foursquare dataset, we find the proposed methodology shows promising performance, in terms of prediction accuracy, comparing with a naive solution based on simple counting the number of co-visitations between users. As for future work, we aim to develop a more comprehensive social connection prediction framework that combines spatio-temporal data with network structure as well as node attribute.

REFERENCES

[1] Lars Backstrom and Jure Leskovec. 2011. Supervised random walks: predicting and recommending links in social networks. In *Proceedings of the fourth ACM international conference on Web search and data mining*. ACM, 635–644.

[2] Eunjoon Cho, Seth A Myers, and Jure Leskovec. 2011. Friendship and mobility: user movement in location-based social networks. In *Proceedings of the 17th ACM SIGKDD conference on knowledge discovery and data mining*. ACM, 1082–1090.

[3] Santo Fortunato. 2010. Community detection in graphs. *Physics reports* 486, 3 (2010), 75–174.

[4] Huiji Gao, Jiliang Tang, and Huan Liu. 2012. Mobile location prediction in spatio-temporal context. In *Nokia mobile data challenge workshop*, Vol. 41. 44.

[5] Neil Zhenqiang Gong, Ameet Talwalkar, Lester Mackey, Ling Huang, Eui Chul Richard Shin, Emil Stefanov, Elaine Runting Shi, and Dawn Song. 2014. Joint link prediction and attribute inference using a social-attribute network. *ACM Transactions on Intelligent Systems and Technology (TIST)* 5, 2 (2014), 27.

[6] Defu Lian, Vincent W Zheng, and Xing Xie. 2013. Collaborative filtering meets next check-in location prediction. In *Proceedings of the 22nd International Conference on World Wide Web*. ACM, 231–232.

[7] Aditya Krishna Menon and Charles Elkan. 2011. Link prediction via matrix factorization. In *Joint european conference on machine learning and knowledge discovery in databases*. Springer, 437–452.

[8] Kurt Miller, Michael I Jordan, and Thomas L Griffiths. 2009. Nonparametric latent feature models for link prediction. In *Advances in neural information processing systems*. 1276–1284.

[9] Anastasios Noulas, Salvatore Scellato, Neal Lathia, and Cecilia Mascolo. 2012. Mining user mobility features for next place prediction in location-based services. In *Data mining (ICDM), IEEE 12th international conference on*. IEEE, 1038–1043.

[10] Ben Taskar Ming-Fai Wong Pieter and Abbeel Daphne Koller. 2003. Link prediction in relational data. (2003).

[11] Salvatore Scellato, Mirco Musolesi, Cecilia Mascolo, Vito Latora, and Andrew T Campbell. 2011. NextPlace: a spatio-temporal prediction framework for pervasive systems. In *International Conference on Pervasive Computing*. Springer, 152–169.

[12] Michael Weiler, Klaus Arthur Schmid, Nikos Mamoulis, and Matthias Renz. 2015. Geo-Social Co-location Mining. In *Second International ACM Workshop on Managing and Mining Enriched Geo-Spatial Data*. ACM, 19–24.

[13] Xiangye Xiao, Yu Zheng, Qiong Luo, and Xing Xie. 2010. Finding similar users using category-based location history. In *ACM SIGSPATIAL*. 442–445.

[14] Dingqi Yang, Daqing Zhang, Longbiao Chen, and Bingqing Qu. 2015. NationTelescope: Monitoring and visualizing large-scale collective behavior in LBSNs. *Journal of Network and Computer Applications* 55 (2015), 170–180.

[15] Jihang Ye, Zhe Zhu, and Hong Cheng. 2013. What's your next move: User activity prediction in location-based social networks. In *Proceedings of the 2013 SIAM International Conference on Data Mining*. SIAM, 171–179.

[16] Zhijun Yin, Manish Gupta, Tim Weninger, and Jiawei Han. 2010. Linkrec: a unified framework for link recommendation with user attributes and graph structure. In *Proceedings of the 19th international conference on World wide web*. ACM, 1211–1212.

[17] Kai Yu, Wei Chu, Shipeng Yu, Volker Tresp, and Zhao Xu. 2006. Stochastic relational models for discriminative link prediction. In *NIPS*. 1553–1560.