

# Link Prediction with Personalized Social Influence

Zepeng Huo,<sup>1</sup> Xiao Huang,<sup>1</sup> Xia Hu<sup>1,2</sup>

<sup>1</sup>Department of Computer Science and Engineering, Texas A&M University

<sup>2</sup>Center for Remote Health Technologies and Systems, Texas A&M Engineering Experiment Station  
{guangzhou92,xhuang,xiahu}@tamu.edu

## Abstract

Link prediction in social networks is to infer the new links likely to be formed next or to reconstruct the links that are currently missing. Other than the pure topological network structures, social networks are often associated with rich information of social activities of users, such as tweeting, retweeting, and replying. Social theories such as social influence indicate that social activities could have potential impacts on the neighbors, and links in social media could be the results of the social influence among users. It motivates us to learn and model social influence among users to tackle the link prediction problem. However, this is a non-trivial task since it is challenging to model heterogeneous social activities. Traditional methods often define universal metrics of social influence for all users, but even for the same activity of a user, the influence towards different neighbors might not be the same. It motivates a personalized learning schema. In information theory, if a time-series signal influences another, then the uncertainty in the latter one will be reduced, given the distribution of the former one. Thus, we are motivated to learn social influence based on the timestamps of social activities. Given the timestamps of each user, we use entropy to measure the reduction of uncertainty of his/her neighbors. The learned social influence is then incorporated into a graph based link prediction model to perform joint learning. Through comprehensive experiments, we demonstrate that the proposed framework can perform better than the state-of-the-art methods on different real-world networks.

## Introduction

As social networks becoming increasingly popular, predicting and reproducing the social network structure draw lots of attentions in recent years (Lichtenwalter, Lussier, and Chawla 2010). Among different problems, link prediction especially directed link prediction is of great interests (Song, Meyer, and Tao 2015). This problem aims to either infer the links that are likely to occur in the near future or reconstruct the existing links that are missing in the current snapshot of the social network. In this paper, we target at the latter one, i.e., link reconstruction problem.

Due to its practical value, link prediction has become an effective computational tool for many real-world applications, such as friend recommendation (Barbieri, Bonchi, and

Manco 2014) and community recommendation (Backstrom et al. 2006). Traditional methods for link prediction can be roughly categorized into two groups. First, some methods use neighbor-based metrics to infer the missing links (Liben-Nowell and Kleinberg 2007), where the similarity functions can be the counts of common neighbors, Jaccard Coefficient of common neighbors, or other variations. Second, some methods employ path-based metrics for link prediction, in which random walk is designed to traverse the paths between two users to calculate the proximity, with one hop or multiple hops (Backstrom and Leskovec 2011). While these existing methods usually focus on the pure topological network structure, users in social media are actually associated with a rich set of social activities, which could be potentially useful in link prediction.

The various kinds of social activities carry abundant information of social media users, where one user's activities are complicatedly intertwined with other user's activities, through social influence (Wasserman and Faust 1994). Social influence in social networks is defined as the phenomenon where we can observe "alteration of an attitude or behavior by one network actor in response to another" (Marsden and Friedkin 1993). Therefore, we could infer the existence of social influence by observing the changing pattern of social activities, which in turn means social influence is not equivalent to social activities, but some kind of quantification of social activities. In social media, the links and dependencies among users could be considered as the results of the social influence taking place. In previous work, social influence has also been demonstrated useful in many applications such as information diffusion study (Bakshy, Karrer, and Adamic 2009) and emotion contagion study (Yang et al. 2016). Thus, we are motivated to collectively model the social influence and network structure to advance link prediction in social media.

However, it is non-trivial to jointly learn and incorporate the heterogeneous social influence and topological structure (Huang, Li, and Hu 2017). Although link prediction on the pure topological network structure has been intensively studied, integrating different types of social activities to infer social influence for link prediction remains an open problem. There are three major challenges as follows. (1) The social activities are heterogeneous. In a social network like Twitter, users can tweet, retweet, reply, and mention oth-

ers. A traditional method is to conduct prolonged feature selection processes on different social activities to extract useful information (Sun and Han 2012), which is not only time-consuming but also domain-specific, since each type of activities needs a specifically designed algorithm. (2) The manifestation of social influence is implicit. Users in social networks will not explain who influences them or how they are influenced. Although a user might receive social influence from people who are followed by him/her, it is implicit to quantify the influence he/she receives. (3) The social influence of each user is neighbor-dependent, i.e., the social influence is not necessarily consistent w.r.t. different neighbors. Traditional social influence methods assign the influence score exclusively to each user, i.e., one user always only has one constant influence score, such as PageRank score and Burt’s network constraint score (Yang et al. 2016). But it can not represent the subtle difference when a user interacts with different neighbors around him/her.

Therefore, in this paper, we propose to use information-theoretic method to calculate the reduction of uncertainty (entropy) of one user’s activities given the knowledge of another one in a pair-wise manner. We use the entropy as the quantification of social influence, which is later used as a regularization for a personalized learning framework, so that the link prediction will benefit from the personalized characteristics of each user. The proposed framework named Personalized Social Influence link prediction (PSI), which learns two low-dimensional representations for each user in a pair-wise social influence schema, i.e., *Source* and *Target*<sup>1</sup>. Thus, each user is no longer represented by a single unchanging influence score, but two vector representations with information of personalized social influence.

The main contributions of this paper are:

- We propose a framework to incorporate social influence model into directed link prediction problem, where social influence can reflect rich set of user activity information.
- Our method gives each user two representations for a personalized social influence schema. It preserves the subtle differences of each user’s interactions with different neighbors, which cannot be reflected by traditional method with a universal influence assignment.
- We propose to use an information-theoretic method to integrate heterogeneous social activities for inferring social influence in a general model without employing any domain-specific feature selection process.
- We conduct experiments on real-world networks to verify that the proposed framework PSI could outperform baselines in link prediction, by preserving the rich information in directed social media.

## Problem Statement

We now introduce the notations and terminologies used in the paper and formally define the link prediction problem in social networks. We use boldface uppercase letters (e.g.  $\mathbf{X}$ )

<sup>1</sup>Source represents one user to follow others, characterizing the features as a follower; Target represents one user being followed by others, characterizing the features as a friend.

to denote matrices and boldface lowercase letters to denote vectors (e.g.  $\mathbf{x}$ ). We use  $\mathbf{X}_i$  to denote the  $i^{\text{th}}$  row of the matrix, and  $\mathbf{X}_{i,j}$  to signify the element in the  $i^{\text{th}}$  row and  $j^{\text{th}}$  column of  $\mathbf{X}$ . The transpose of  $\mathbf{X}$  is represented as  $\mathbf{X}^T$ . The  $\ell_2$ -norm of a vector is represented by  $\|\cdot\|_2$ , and the Frobenius norm of a matrix is denoted as  $\|\cdot\|_F$ .

Let  $\mathcal{G} = \{\mathcal{U}, \mathcal{E}\}$  be a directed network, where  $\mathcal{U}$  indicates a set of  $N$  users  $\{u_1, u_2, \dots, u_N\}$  and  $\mathcal{E} \subseteq \mathcal{U} \times \mathcal{U}$  indicates the corresponding edge set. We denote a directed edge from  $u_i$  to  $u_j$  as  $(i, j) \in \mathcal{E}$ . Let  $\mathcal{A}$  be a set of  $N$  sequences of timestamps. For each user  $u_i$ , the timestamp sequence  $\mathcal{A}^{(i)} = \{t_{i,1}, t_{i,2}, \dots\}$  records the occurrence time of all his/her online social activities. The time intervals of these activities could vary from seconds to months. Based on the terminologies defined above, we formally define the link prediction problem in social networks as follows.

*Given a directed graph  $\mathcal{G}$  associated with a set of edges  $\mathcal{E}$  and a set of timestamp sequences  $\mathcal{A}$  that records the occurrence of all users’ social activities, we aim to predict the probability of having a directed edge from any user  $u_i$  to any other user  $u_j$ , jointly based on the network topological structure in  $\mathcal{G}$  and social activity information in  $\mathcal{A}$ .*

## Prediction with Personalized Social Influence

To jointly model the topological structure and social activity information, we propose a link prediction framework named Personalized Social Influence (PSI). The main idea is to learn two low-dimensional vector representations for each user  $u_i$ , i.e., Source representation  $\mathbf{S}_i \in \mathbb{R}^{1 \times d}$  and Target representation  $\mathbf{T}_i \in \mathbb{R}^{1 \times d}$ , such that all the social influence among linked users is well preserved. The influence is directional and its strength depends on the social activity information in  $\mathcal{A}$ . Thus, we have two vector representations for each user, aiming to represent its roles in being affected and giving impacts respectively. The proposed framework PSI could be separated into three major components as follows. First, it quantifies the strength of social influence based on the underlying patterns in the occurrence time of user activities. Second, it jointly embeds the directional network structure and the learned social activity information into two low-dimensional representations  $\mathbf{S}$  and  $\mathbf{T}$ . Third, it accelerates the optimization via the Negative Sampling technique (Mikolov et al. 2013). As a result, we could predict the probability of having a link from any user  $u_i$  to any other user  $u_j$  based on  $\mathbf{S}_i \mathbf{T}_j$ .

## Social influence quantification

We now introduce how to model the social activity information. In this paper, we focus on the timestamp information, and the main reason is that other types of information of social activities could be heterogeneous and usually text-based, whose processing is computationally expensive. To model the timestamp information, an intuitive solution is to consider the timestamps as features and stack up every user’s  $\mathcal{A}$  into a unified feature matrix. However, this solution is not applicable to our problem since the learned feature matrix would be extremely sparse, due to that some users’ may only have a few activity timestamps in their sequence  $\mathcal{A}$ .

To quantify the social influence based on user activity information, we first personalize the timestamp sequence set  $\mathcal{A}$  as a matrix  $\mathbf{A}^{(i)}$  for each user  $u_i$ , so each user will have a unique matrix, which is later used in personalized learning.

**Timestamp sequence modeling** For each user  $u_i$ , we have a timestamp sequence  $\{t_{i,1}, t_{i,2}, \dots\}$  that records the occurrence time of all his/her activities. Since the active periods of different users are quite diverse, we define a personalized time interval  $\Delta t^i$  for user  $u_i$  as follows.

$$\Delta t^i = (t_{max} - t_{min})/M, \quad (1)$$

where  $t_{min}$  and  $t_{max}$  denote the timestamps of the first and last activities of  $u_i$ , and  $M$  is a predefined maximum number of time intervals. We define the activity frequency of  $u_i$  in the  $m^{\text{th}}$  interval as  $\mathbf{A}_{i,m}^{(i)}$ . Note the activity matrix has been personalized based on the time interval  $\Delta t^i$  as  $\mathbf{A}^{(i)}$ , which will be used to calculate the personalized social influence of user  $u_i$ . Similarly  $\mathbf{A}_{i,m}^{(j)}$  denotes  $u_i$ 's activities at  $m^{\text{th}}$  time interval, which is derived from  $u_j$ 's personalized interval. Then the sequence set  $\mathcal{A}$  is personalized as a matrix  $\mathbf{A}^{(i)} \in \mathbb{R}^{N \times M}$  for each user. It should be noted that, for different users, their activity time intervals are different, varying from seconds to months. Thus, we have  $N$  different personalized social activity matrices  $\{\mathbf{A}^{(i)}\}$ , for  $i = 1, 2, \dots, N$ .

**Metric of social influence** We now focus on a pair of linked users, with a directional edge  $(i, j)$  denoting  $u_i$  following  $u_j$ , where the influence is actually from  $u_j$  to  $u_i$ . We quantify this social influence by measuring the reduction of uncertainty of  $\mathbf{A}_i^{(i)}$  given the knowledge of  $\mathbf{A}_j^{(i)}$ . It should be noted that we use the personalized time interval of the user being influenced. We infer the influence by comparing two scenarios as follows.

(1) We construct the probability of  $u_i$  being active given his/her own activities. By considering vector  $\mathbf{A}_i^{(i)}$  as a Markov chain with  $M$  variables, we could infer the activity frequency  $\mathbf{A}_{i,m+1}^{(i)}$  based on the historical record  $\mathbf{A}_{i,m}^{(i)}$ . We denote the probability of  $u_i$  being active at  $m+1$  as,

$$\mathbf{p}_1 = \left\{ \mathbb{P}(\mathbf{A}_{i,m+1}^{(i)} \neq 0 | \mathbf{A}_{i,m}^{(i)}) \right\}, \text{ for } m = 1, \dots, M-1, \quad (2)$$

where  $\mathbf{p}_1$  is a probability distribution with  $M-1$  probability values. The intuition of  $\mathbf{p}_1$  is the probabilities that we can use  $u_i$ 's own history to predict his/her future activities.

(2) We construct the possibility of  $u_i$  being active given his/her own and  $u_j$ 's history activities. This probability could often be reflected as the dependency from activity frequency  $\mathbf{A}_i^{(i)}$  to  $\mathbf{A}_j^{(i)}$  (Ver Steeg and Galstyan 2012). Mathematically, the probability is defined as follows,

$$\mathbf{p}_2 = \left\{ \mathbb{P}(\mathbf{A}_{i,m+1}^{(i)} \neq 0 | \mathbf{A}_{i,m}^{(i)}, \mathbf{A}_{j,m}^{(i)}) \right\}, \text{ for } m = 1, \dots, M-1. \quad (3)$$

Thus, we get another probability distribution  $\mathbf{p}_2$ .

It should be noted that we only consider the influence among adjacent time intervals, e.g.  $m$  and  $m+1$ , since the influence is often stronger than the non-adjacent one. Motivated by the study of influence flow (Ver Steeg and Galstyan

2013), we employ the entropy reduction of  $\mathbf{p}_1$  from  $\mathbf{p}_2$  as a metric of social influence from  $u_j$  to  $u_i$ , i.e.,

$$\begin{aligned} I_{j \rightarrow i} &\triangleq H(\mathbf{p}_1) - H(\mathbf{p}_2) \\ &= - \sum_{m=1}^{M-1} \mathbb{P}(\mathbf{A}_{i,m+1}^{(i)} \neq 0 | \mathbf{A}_{i,m}^{(i)}) \log \mathbb{P}(\mathbf{A}_{i,m+1}^{(i)} \neq 0 | \mathbf{A}_{i,m}^{(i)}) \\ &\quad + \sum_{m=1}^{M-1} \mathbb{P}(\mathbf{A}_{i,m+1}^{(i)} \neq 0 | \mathbf{A}_{i,m}^{(i)}, \mathbf{A}_{j,m}^{(i)}) \log \mathbb{P}(\mathbf{A}_{i,m+1}^{(i)} \neq 0 | \mathbf{A}_{i,m}^{(i)}, \mathbf{A}_{j,m}^{(i)}), \end{aligned} \quad (4)$$

where  $H(\mathbf{p}_1)$  and  $H(\mathbf{p}_2)$  denote the entropy of distributions  $\mathbf{p}_1$  and  $\mathbf{p}_2$ . The key idea of metric  $I_{j \rightarrow i}$  is that, when  $u_j$  has influence on  $u_i$ , then  $\mathbf{A}_j^{(i)}$  could reduce the uncertainty of predicting  $\mathbf{A}_i^{(i)}$ . Therefore, we have mathematically calculated social influence in a personalized schema.

**Computation of  $\mathbf{p}_1$  and  $\mathbf{p}_2$**  It should be noted that we need to use a set of probability values to calculate those two aforementioned entropy terms, which requires that  $\mathbf{p}_1$  and  $\mathbf{p}_2$  are two sets of posterior probabilities. First, for calculation of  $\mathbf{p}_1$ , we employ a logistic regression model to calculate the conditional probability of  $u_i$  being active given his/her own history,

$$\mathbb{P}(\mathbf{A}_{i,m+1}^{(i)} \neq 0 | \mathbf{A}_{i,m}^{(i)}) = \frac{1}{1 + e^{-\alpha_0 - \alpha_1 f(\mathbf{A}_{i,m}^{(i)})}}, \quad (5)$$

where  $\alpha_0$  and  $\alpha_1$  are coefficients learned from the entire record. Specifically, we employ the sequence  $\{\mathbf{A}_{i,1}^{(i)}, \mathbf{A}_{i,2}^{(i)}, \dots, \mathbf{A}_{i,M-1}^{(i)}\}$  as  $M-1$  training inputs, and the binary sequence  $\{\text{sgn}(\mathbf{A}_{i,2}^{(i)}), \text{sgn}(\mathbf{A}_{i,3}^{(i)}), \dots, \text{sgn}(\mathbf{A}_{i,M}^{(i)})\}$  as corresponding labels, where function  $\text{sgn}(\cdot)$  is the sign function. We also employ a discount function  $f(x)$  in Eq. (5) for each activity frequency, which is defined as follows.

$$f(x) = \begin{cases} x, & \text{if } x \leq 2, \\ 1 + \lceil \log(1+x) \rceil, & \text{o/w.} \end{cases} \quad (6)$$

The basic idea behind  $f(x)$  is that as more activities occurred during only one time interval  $\Delta t^i$ , the number of activities actually seen by people would not increase linearly. For instance, in Twitter, the activity frequency could be the number of tweets that a user  $u_j$  has posted during one time interval  $\Delta t^i$ . If  $u_j$  posts a large number of tweets in a short period, his/her follower  $u_i$  may not check all of them (Quinn, Kiyavash, and Coleman 2015). Thus we employ a discount function  $f(x)$  to estimate the actual number of activities that could be perceived by  $u_i$ .

Similarly, we could calculate  $\mathbf{p}_2$ , with pre-trained coefficients  $\alpha_0$ ,  $\alpha_1$ , and  $\alpha_2$ , i.e.,

$$\mathbb{P}(\mathbf{A}_{i,m+1}^{(i)} \neq 0 | \mathbf{A}_{i,m}^{(i)}, \mathbf{A}_{j,m}^{(i)}) = \frac{1}{1 + e^{-\alpha_0 - \alpha_1 f(\mathbf{A}_{i,m}^{(i)}) - \alpha_2 f(\mathbf{A}_{j,m}^{(i)})}}. \quad (7)$$

By substituting  $\mathbf{p}_1$  and  $\mathbf{p}_2$  into Eq. (4), we can calculate the reduction of uncertainty of  $u_i$ 's activities given the knowledge of  $u_j$ 's activities, which is defined as the personalized social influence in this paper.

## Jointly modeling social activity and network

Since each user  $u_i$  could both be influenced by others and give influence to others, we employ two vector representations to represent  $u_i$ , i.e., Source representation  $\mathbf{S}_i$  and Target representation  $\mathbf{T}_i$ . In such way, we could predict the probability of having a link from any user  $u_i$  to any other user  $u_j$  based on  $\mathbf{S}_i\mathbf{T}_j$ . The main goal is to make sure  $\mathbf{S}_i\mathbf{T}_j > \mathbf{S}_i\mathbf{T}_n$  for any pair of existing edge  $(i, j) \in \mathcal{E}$  and non-existent edge  $(i, n) \notin \mathcal{E}$ , which is essentially the network structure information.

To jointly model the topological structure and social activity information, our key idea is to learn informative  $\mathbf{S}$  and  $\mathbf{T}$ , such that the aforementioned probability of a user following his/her true friend ( $\mathbf{S}_i\mathbf{T}_j$ ) is larger than a random user ( $\mathbf{S}_i\mathbf{T}_n$ ). And the difference between them is proportional to the social influence metric.

We now define an empirical probability of having a directed edge from user  $u_i$  to  $u_j$  as follows. The main idea is to calculate the amount of influence solely from  $u_j$  towards  $u_i$ , comparing with all other potentially influential users.

$$\hat{p}(u_j|u_i) = I_{j \rightarrow i} / d_i^{\text{out}}, \quad (8)$$

where  $d_i^{\text{out}} = \sum_{(i,n) \in \hat{\mathcal{E}}} I_{n \rightarrow i}$  is the out-degree of user  $u_i$ .

Note we define a new set of edges  $\hat{\mathcal{E}}$  that only includes node pairs with significant influence, i.e., we only consider node pairs with social influence greater than a threshold,

$$I_{j \rightarrow i} > \frac{\log_2(\sum_{m=1}^M \mathbf{A}_{i,m})}{2 \sum_{m=1}^M \mathbf{A}_{i,m}}. \quad (9)$$

The threshold is motivated by Minimum Description Length penalty (MDL) (Grünwald 2007). The main reason for applying a threshold here is that more active users will be more likely to overfit the model, but less active users may not be learned properly.

We employ the softmax function to calculate the probability of having a directed edge from user  $u_i$  to  $u_j$  through their representations (Tang et al. 2015),

$$p(u_j|u_i) = \frac{e^{\mathbf{S}_i\mathbf{T}_j^\top}}{\sum_{n=1}^N e^{\mathbf{S}_i\mathbf{T}_n^\top}}. \quad (10)$$

The key idea is to determine the linking probability based on the contribution of user  $u_i$  to  $u_j$  comparing with all other users in the pool.

Therefore, by minimizing the Kullback–Leibler divergence of  $p(u_j|u_i)$  and its empirical counterpart  $\hat{p}(u_j|u_i)$ , we can get the objective function:

$$\begin{aligned} \min \mathcal{J}_1 &= \sum_{(i,j) \in \hat{\mathcal{E}}} d_i^{\text{out}} D_{\text{KL}}(\hat{p}(u_j|u_i) || p(u_j|u_i)) \\ &= - \sum_{(i,j) \in \hat{\mathcal{E}}} I_{j \rightarrow i} \log p(u_j|u_i) + \frac{I_{j \rightarrow i}}{d_i^{\text{out}}} \log \frac{I_{j \rightarrow i}}{d_i^{\text{out}}}, \end{aligned} \quad (11)$$

where  $d_i^{\text{out}}$  represents the prestige of user  $u_i$  in the network, which is defined before. After ignoring the constant, the objective can be simply rewritten as:

$$\max \hat{\mathcal{J}}_1 = \sum_{(i,j) \in \hat{\mathcal{E}}} I_{j \rightarrow i} \log p(u_j|u_i). \quad (12)$$

Since we want to minimize the divergence, it is equivalent to maximizing the objective by omitting the negative sign.

## Acceleration via negative sampling

We can see optimizing  $\hat{\mathcal{J}}_1$  is computationally expensive since by calculating one softmax as Eq. (10), every pair of users needs to compare with all the users. So before we optimize  $\hat{\mathcal{J}}_1$ , we would like to rewrite Eq. (10) as follows.

$$p(u_j|u_i) = \frac{1}{1 + \sum_{n=1, n \neq j}^N e^{-(\mathbf{S}_i\mathbf{T}_j^\top - \mathbf{S}_i\mathbf{T}_n^\top)}}. \quad (13)$$

It follows the format of Eq. (10), just by dividing both numerator and denominator by  $e^{\mathbf{S}_i\mathbf{T}_j^\top}$ . Note the form of sigmoid function  $\sigma(x) = 1/(1 + e^{-x})$ , we can define a new conditional probability in the similar form of Eq. (13):

$$p(u_j > u_n|u_i) = \sigma(\mathbf{S}_i\mathbf{T}_j^\top - \mathbf{S}_i\mathbf{T}_n^\top). \quad (14)$$

The above conditional probability can be interpreted as instead of directly optimizing  $\hat{\mathcal{J}}_1$  over all users, we update Eq. (14) with respect to a small set of noise samples in  $\mathcal{U} \setminus j$ , where an individual sample is denoted as  $u_n$  (Gui et al. 2016). It can be easily verified that:

$$p(u_j|u_i) > \prod_{u_n \in \mathcal{U} \setminus j} p(u_j > u_n|u_i). \quad (15)$$

Therefore, instead of optimizing  $\hat{\mathcal{J}}_1$ , we can optimize a tight lower bound of  $p(u_j|u_i)$  in  $\hat{\mathcal{J}}_1$ . So if we combine Eqs. (14) and (15) and put them back to objective  $\hat{\mathcal{J}}_1$  in Eq. (12), we can get a new objective function:

$$\max \mathcal{J}_2 = \sum_{(i,j) \in \hat{\mathcal{E}}} I_{j \rightarrow i} \sum_{u_n \in \mathcal{U} \setminus j} \log \sigma(\mathbf{S}_i\mathbf{T}_j^\top - \mathbf{S}_i\mathbf{T}_n^\top). \quad (16)$$

It should be noted that the  $I_{j \rightarrow i}$  needs to satisfy Eq. (9). To accelerate the learning process, here we adopted Negative Sampling in (Mikolov et al. 2013). All the negative samples will be drawn from a noise distribution. So the probability part can be further rewritten as:

$$\begin{aligned} &\sum_{u_n \in \mathcal{U} \setminus j} \log \sigma(\mathbf{S}_i\mathbf{T}_j^\top - \mathbf{S}_i\mathbf{T}_n^\top) \\ &\propto \sum_n^K E_{u_n \sim P_n(u)} \log \sigma(\mathbf{S}_i\mathbf{T}_j^\top - \mathbf{S}_i\mathbf{T}_n^\top). \end{aligned} \quad (17)$$

where  $K$  is the number of negative instances, sampled from noise distribution of  $P_n(u) \propto d_u^{3/4}$ , and  $d_u$  is the out-degree of user  $u$ . Thus, there is no need to go through all users to get the conditional probability, but just fewer noise samples.

We can then write the final objective function in a unified form, incorporating network structure and social influence with acceleration learning schema:

$$\begin{aligned} \max \mathcal{J} &= \sum_{(i,j) \in \hat{\mathcal{E}}} I_{j \rightarrow i} \sum_n^K E_{u_n \sim P_n(u)} \log \sigma(\mathbf{S}_i\mathbf{T}_j^\top - \mathbf{S}_i\mathbf{T}_n^\top) \\ &\quad - \frac{\beta_1}{2} \|\mathbf{S}\|_F^2 - \frac{\beta_2}{2} \|\mathbf{T}\|_F^2, \end{aligned} \quad (18)$$



The last two terms  $\|\mathbf{S}\|_F^2$  and  $\|\mathbf{T}\|_F^2$  are employed to avoid overfitting.  $\beta_1$  and  $\beta_2$  are the regularization coefficients.

### Computations of $\mathbf{S}$ and $\mathbf{T}$

Next we will derive the update rules of each model parameter  $\Theta = \mathbf{S}$  or  $\mathbf{T}$ . In each iteration, we can update model parameter according to asynchronous stochastic gradient descent, which is a fast optimization algorithm in many machine learning applications. For each model parameter  $\Theta$ , we derive the gradient from Eq. (18) as follows.

$$\frac{\partial \mathcal{J}}{\partial \Theta} = \sum_{(i,j) \in \mathcal{E}} I_{j \rightarrow i} \sum_n^K E_{u_n \sim P_n(u)} \frac{\partial \mathcal{L}(u_i, u_j, u_n)}{\partial \Theta} - \beta_{\Theta} \Theta, \quad (19)$$

where we define  $\mathcal{L}(u_i, u_j, u_n) = \log \sigma(\mathbf{S}_i \mathbf{T}_j^\top - \mathbf{S}_i \mathbf{T}_n^\top)$ . Therefore, we only need to calculate the derivative of  $\mathcal{L}(u_i, u_j, u_n)$  w.r.t. each  $\Theta = \mathbf{S}$  or  $\mathbf{T}$  since other calculation is the same for each model parameter. So for user Source representation  $\mathbf{S}$ , we have,

$$\frac{\partial \mathcal{L}(u_i, u_j, u_n)}{\partial \mathbf{S}_i} = \epsilon \frac{\partial \sigma(\mathbf{S}_i \mathbf{T}_j^\top - \mathbf{S}_i \mathbf{T}_n^\top)}{\partial \mathbf{S}_i}, \quad (20)$$

where  $\epsilon$  is defined as  $\epsilon = 1/\sigma(\mathbf{S}_i \mathbf{T}_j^\top - \mathbf{S}_i \mathbf{T}_n^\top)$ . As for user Target representation  $\mathbf{T}$ , we have,

$$\begin{aligned} \frac{\partial \mathcal{L}(u_i, u_j, u_n)}{\partial \mathbf{T}_j} &= \epsilon \frac{\partial \sigma(\mathbf{S}_i \mathbf{T}_j^\top - \mathbf{S}_i \mathbf{T}_n^\top)}{\partial \mathbf{T}_j}, \\ \frac{\partial \mathcal{L}(u_i, u_j, u_n)}{\partial \mathbf{T}_n} &= -\epsilon \frac{\partial \sigma(\mathbf{S}_i \mathbf{T}_j^\top - \mathbf{S}_i \mathbf{T}_n^\top)}{\partial \mathbf{T}_n}. \end{aligned} \quad (21)$$

For simplicity, we omit the writing of regularization term  $\beta_{\Theta} \Theta$  of each model parameter.

## Experiment

In this section, we empirically evaluate our framework PSI by comparing it with the state-of-the-art methods. We aim to answer two questions: (1) What is the impact of learning social influence on performing link prediction? (2) How effective is our method in modeling social activity information, especially when it is sparse?

### Datasets

We use two publicly available datasets in our experiment: URL Twitter dataset (Hodas and Lerman 2014) and Higgs Twitter dataset (Leskovec and Krevl 2014). These two datasets are suitable for our experiment because they both have complete friendship information and users' activities. The details of these two datasets are shown in Table 1.

### Baseline methods

We compare our framework PSI in the directed link prediction problem with the state-of-the-art methods, which could be separated into three categories. First, to investigate the effectiveness of our proposed framework PSI in learning the network structure, we compare it with a method that only learns from the social activities, i.e., R&M. Second,

Dataset	URL dataset	Higgs dataset
# of users	736,930	456,626
# of user activities	2,859,764	563,069
# of directed links	36,743,448	14,855,842

Table 1: Statistics of Experimental Dataset

to study the impact of considering social activity information for link prediction problem, we compare PSI with baselines that only consider network structure, i.e., CN, BPR-MF, and ELLR. Third, to demonstrate the effectiveness of PSI in jointly learning social activities and network structure, we include baseline methods TI and SRW. The details of baselines are shown as follows.

- R&M (Cha et al. 2010): The directed links are inferred by the counts of retweet and mention of each pair of users.
- CN (Liben-Nowell and Kleinberg 2007): The Common Neighbor method is widely adopted for link prediction problem, due to its simplicity for implementation.
- BPR-MF (Rendle et al. 2009): It is the Bayesian Personalized Ranking in matrix factorization framework for predicting the links between users.
- ELLR (Song, Meyer, and Tao 2015): It uses a generalized AUC for an Efficient Latent Link Recommendation.
- TI (Pálovics et al. 2014): The method exploits Temporal Influence for link prediction by using matrix factorization. The temporal information is based on time delay of each pair of users, where smaller delay means higher influence.
- SRW (Backstrom and Leskovec 2011): The Supervised Random Walks method learns the edge weights to let the random walker more likely to traverse nodes that have edges with current nodes. We use concatenation of two users' activities record as an edge vector between them.

### Experimental setup and evaluation metrics

We conduct the experiment from two aspects which are well established in link prediction problems to test the algorithm performance, i.e., *pair-wise* accuracy and *list-wise* accuracy (Zhang, Wang, and Feng 2013). Note that we will use all user's timestamps in the experiment instead of only those before the formations of the link because we target at the problem of reconstruction of the links.

First, to test pair-wise accuracy, each test instance is a tuple with three users, i.e., a user, his/her true friend (positive instance) and a random user (negative instance). We aim to measure whether the algorithm can distinguish a positive link from the negative one, i.e., pair-wise accuracy. And we average the accuracy of all the test instances to have the final pair-wise accuracy. The metric we adopted is AUC (Area Under the Curve). In terms of separating training and testing set, we randomly select different subset of tuples from dataset (i.e., 10%, 20%, 40%, 60%) to train the model. For each fraction of training set, all the remaining instances will be used as test set.

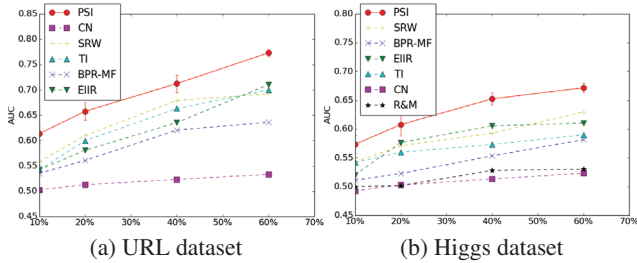


Figure 1: Area Under the Curve (AUC)

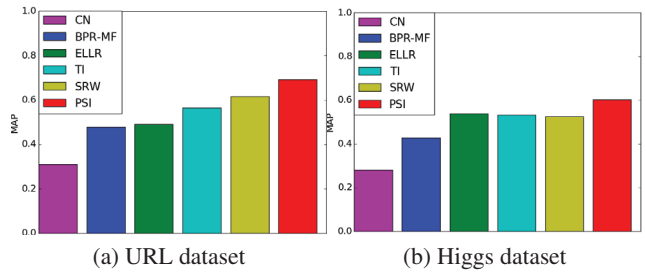


Figure 3: Mean Average Precision (MAP)

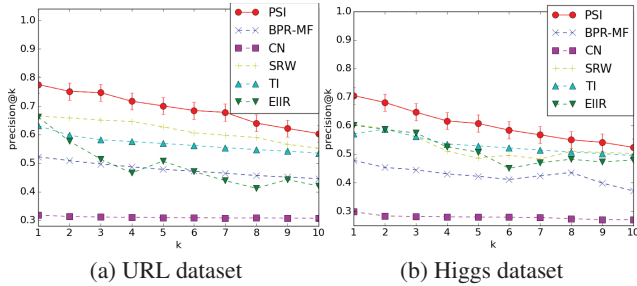


Figure 2: Precision@k

Second, the list-wise accuracy measures the portion of true friends in a ranked list of recommended friends returned by the algorithms, where better algorithm intuitively will give true friends higher rank in the list. So we adopt Precision@k to measure the accuracy in the ranked list of users in different positions, and MAP (Mean Average Precision). We evaluate the precision at the first ten positions of the ranked list. In terms of separating training and testing set, all the directed links are randomly divided into two groups, where 60% is for training the model, and 40% is for testing the list-wise accuracy.

### Hyper-parameter discussion

(1) Learning rate and regularization parameters. We conduct a grid search over candidate set of learning rates and regularization parameters. So we set the learning rate as 0.01 and regularization parameters of representations as 0.025.

(2) Likelihood function coefficients. Numerically, for the coefficients  $\alpha = \{\alpha_0, \alpha_1, \alpha_2\}$  in Eqs. (5) and (7), a negative coefficient was more likely because of over-fitting than the situation that user  $u_j$  will suppress user  $u_i$ 's activities. Therefore, if any coefficient happens to be negative in likelihood function, it will be rejected.

### Experimental results

We evaluate the results based on the questions that we asked at the beginning of this section, by comparing the proposed method PSI with the baseline methods. Next we will discuss the experimental results in detail.

**1. Effectiveness of jointly learning** We now answer the first question, i.e., how effective could PSI jointly learn so-

cial activity and network structure information compared to other methods. We investigate this question by looking into the two accuracy metrics.

*A. Pair-wise accuracy.* The results of pair-wise accuracy in terms of AUC metric are shown in Figure 1. Note that URL dataset does not have information of retweet and mention, so we only conduct R&M on Higgs dataset with AUC metric. We have several observations as follows.

(1) In general, the methods which consider both user activity and network structure information outperformed the methods that only consider one information source. For instance, SRW achieves 18.7% and 20.3% gain over R&M and CN respectively with 60% training set in Higgs dataset. However, our method further achieves 6.68% gain over SRW, indicating that the PSI can better integrate social activities with network structure information.

(2) The proposed method PSI has better performance with smaller training set. For example, in URL dataset, with only 40% of training set our method has higher accuracy than most of the baselines with 60% training set. In Higgs dataset, our method even only requires 20% of training to outperform most of the baselines with 60% training set.

(3) Methods generally have better performance in URL dataset than in Higgs dataset, which could be explained by the sparsity of Higgs dataset in terms both following relationships and user activities.

*B. List-wise accuracy.* We now investigate the results of list-wise accuracy by using the Precision@k and MAP as shown in Figures 2 and 3. We draw several observations as follows.

(1) Methods that combine both network structure and user activities generally outperform the methods that consider only one. But again the improvement is more salient in URL dataset than in Higgs dataset.

(2) Our method performs relatively stable in different positions in Precision@k metric in Figure 2, while some of the baselines did not, such as ELLR in URL dataset in positions 4 to 9, and BPR-MF in Higgs in positions 6 to 9. We can see these two methods have drastic changes in those positions. We infer that without using social activities as the extra information to regularize the model, those methods with only network structure information will be largely affected by the noisy instances in recommending a list of users.

(3) The social activity information affects the models in an evident way. As shown in Figure 3, in URL dataset, ELLR performs worse than TI and SRW w.r.t. MAP metric. How-

	10% (gain)	25% (gain)	50% (gain)	100% (gain)
TI	0.4813 (N.A.)	0.5187 (N.A.)	0.5323 (N.A.)	0.5647 (N.A.)
SRW	0.4998 (+3.87%)	0.5453 (+5.12%)	0.5738 (+7.79%)	0.6165 (+9.15%)
PSI	<b>0.5438 (+12.91%)</b>	<b>0.5965 (+14.97%)</b>	<b>0.6333 (+18.94%)</b>	<b>0.6919 (+22.51%)</b>

Table 2: Model Sensitivity on User Activity Size of URL dataset

	10% (gain)	25% (gain)	50% (gain)	100% (gain)
TI	0.3925 (N.A.)	0.4364 (N.A.)	0.4854 (N.A.)	0.5327 (N.A.)
SRW	0.4303 (+9.63%)	0.4455 (+2.08%)	0.4922 (+1.40%)	0.5250 (-1.44%)
PSI	<b>0.4538 (+15.61%)</b>	<b>0.5053 (+15.78%)</b>	<b>0.5322 (+9.64%)</b>	<b>0.6026 (+13.12%)</b>

Table 3: Model Sensitivity on User Activity Size of Higgs dataset

ever, in Higgs dataset where the social activities are relatively sparse, EIR even performs slightly better than these two methods, indicating these two methods cannot properly learn from sparse data. But our method consistently outperforms those methods, meaning that it can better learn useful information from sparse social activities.

**2. Impact of sparsity of social activities** We now answer the second question, i.e., how effective of our method in modeling social activity information, especially when it is sparse. We will compare the methods that consider both network structure and user activity information (TI and SRW) with our proposed method. We first sequentially (time sequence from oldest to newest) sample different portions of each user’s activities, e.g., if we sample 10% of user activities, it means only the first 10% of social activities are included. We use MAP as the evaluation metric. The results are shown in Table 2 and 3. Each column represents how much user activities are used, and ‘gain’ means the percentage improvement of the methods as compared to TI. The result shows the robustness of our method with sparse user activities. In URL dataset, compared with TI and SRW, when only 10% of user activities are considered, our method already outperforms TI and SRW by 12.91% and 8.80%, respectively. When we include all social activity information, our method can achieve 22.51% and 12.23% gain at maximum over TI and SRW. In Higgs dataset, we can observe the similar outcome, for only 10% activities considered, our method outperforms TI and SRW by 15.61% and 5.46%, and the maximum gain is achieved in 25% training set. In summary, our method can properly learn useful information from sparse social activities better than baselines methods, and thus performs better in link prediction.

## Related Work

For link prediction in social networks, people have been drawn attentions to it after the innovative work by Liben-Nowell and Kleinberg (2007). In general, most of the approaches are designed to calculate different kinds of proximity metrics on social networks as prediction features (Chen,

Li, and Huang 2005; Lichtenwalter, Lussier, and Chawla 2010), where the learning framework includes supervised (Al Hasan et al. 2006) and unsupervised (Liben-Nowell and Kleinberg 2007). In social networks, directed links are of great interest in many studies. Valverde-Rebaza and Lopes (2013) proposed to combine community information with topology to predict links in a directed and asymmetric social network. Hopcroft and Tang (2011) studied the reciprocal relationship prediction in directed social networks.

Methods with social influence model are well-studied in various domains, such as sociology and marketing literature (Granovetter 1978; Goldenberg, Libai, and Muller 2001). The social influence study often focuses on finding the most influential nodes, which has been applied to different applications, such as emotional contagion (Yang et al. 2016), and information diffusion by using IC (Independent Cascade) model (Kempe, Kleinberg, and Tardos 2003).

## Conclusions

In this paper, we advance the link prediction in social networks by considering the social influence. While link prediction has been intensively studied with pure network structures, we demonstrate that by modeling and incorporating social influence information into the network topological structure, our method could perform better in link prediction. The quantification of social influence is learned from social activities through information-theoretic method, and the personalized social influence is further preserved in the Source and Target representations. On real-world datasets, our framework PSI outperforms the state-of-the-art methods, indicating its effectiveness in jointly learning the social activity information and topological structure.

## Acknowledgments

This work is, in part, supported by DARPA (#N66001-17-2-4031) and NSF (#IIS-1657196 and #IIS-1718840). The views, opinions, and/or findings expressed are those of the author(s) and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government.

## References

- Al Hasan, M.; Chaoji, V.; Salem, S.; and Zaki, M. 2006. Link prediction using supervised learning. In *SDM06: workshop on link analysis, counter-terrorism and security*.
- Backstrom, L., and Leskovec, J. 2011. Supervised random walks: predicting and recommending links in social networks. In *ACM International Conference on Web Search and Data Mining*, 635–644.
- Backstrom, L.; Huttenlocher, D.; Kleinberg, J.; and Lan, X. 2006. Group formation in large social networks: membership, growth, and evolution. In *ACM International Conference on Knowledge Discovery and Data Mining*, 44–54.
- Bakshy, E.; Karrer, B.; and Adamic, L. A. 2009. Social influence and the diffusion of user-created content. In *ACM conference on Electronic commerce*, 325–334.
- Barbieri, N.; Bonchi, F.; and Manco, G. 2014. Who to follow and why: link prediction with explanations. In *ACM International conference on Knowledge Discovery and Data Mining*, 1266–1275.
- Cha, M.; Haddadi, H.; Benevenuto, F.; and Gummadi, P. K. 2010. Measuring user influence in twitter: The million follower fallacy. In *International Conference on Web and Social Media*, 10–17.
- Chen, H.; Li, X.; and Huang, Z. 2005. Link prediction approach to collaborative filtering. In *ACM/IEEE-CS Joint Conference on Digital Libraries*, 141–142.
- Goldenberg, J.; Libai, B.; and Muller, E. 2001. Talk of the network: A complex systems look at the underlying process of word-of-mouth. *Marketing letters* 12(3):211–223.
- Granovetter, M. 1978. Threshold models of collective behavior. *American journal of sociology* 83(6):1420–1443.
- Grünwald, P. D. 2007. *The minimum description length principle*. MIT press.
- Gui, H.; Liu, J.; Tao, F.; Jiang, M.; Norick, B.; and Han, J. 2016. Large-scale embedding learning in heterogeneous event data. In *IEEE International Conference on Data Mining*, 907–912.
- Hodas, N. O., and Lerman, K. 2014. The simple rules of social contagion. *Scientific reports* 4.
- Hopcroft, J.; Lou, T.; and Tang, J. 2011. Who will follow you back?: reciprocal relationship prediction. In *ACM International Conference on Information and Knowledge Management*, 1137–1146.
- Huang, X.; Li, J.; and Hu, X. 2017. Label informed attributed network embedding. In *ACM International Conference on Web Search and Data Mining*, 731–739.
- Kempe, D.; Kleinberg, J.; and Tardos, É. 2003. Maximizing the spread of influence through a social network. In *ACM International Conference on Knowledge Discovery and Data Mining*, 137–146.
- Leskovec, J., and Krevl, A. 2014. SNAP Datasets: Stanford large network dataset collection. <http://snap.stanford.edu/data>.
- Liben-Nowell, D., and Kleinberg, J. 2007. The link-prediction problem for social networks. *Journal of the Association for Information Science and Technology* 58(7):1019–1031.
- Lichtenwalter, R. N.; Lussier, J. T.; and Chawla, N. V. 2010. New perspectives and methods in link prediction. In *ACM International conference on Knowledge Discovery and Data Mining*, 243–252.
- Marsden, P. V., and Friedkin, N. E. 1993. Network studies of social influence. *Sociological Methods & Research* 22(1):127–151.
- Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G. S.; and Dean, J. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, 3111–3119.
- Pálovics, R.; Benczúr, A. A.; Kocsis, L.; Kiss, T.; and Frigó, E. 2014. Exploiting temporal influence in online recommendation. In *PACM Conference on Recommender systems*, 273–280.
- Quinn, C. J.; Kiyavash, N.; and Coleman, T. P. 2015. Directed information graphs. *IEEE Transactions on Information Theory* 61(12):6887–6909.
- Rendle, S.; Freudenthaler, C.; Gantner, Z.; and Schmidt-Thieme, L. 2009. Bpr: Bayesian personalized ranking from implicit feedback. In *Conference on Uncertainty in Artificial Intelligence*, 452–461.
- Song, D.; Meyer, D. A.; and Tao, D. 2015. Efficient latent link recommendation in signed networks. In *ACM International Conference on Knowledge Discovery and Data Mining*, 1105–1114.
- Sun, Y., and Han, J. 2012. Mining heterogeneous information networks: principles and methodologies. *Synthesis Lectures on Data Mining and Knowledge Discovery* 3(2):1–159.
- Tang, J.; Qu, M.; Wang, M.; Zhang, M.; Yan, J.; and Mei, Q. 2015. Line: Large-scale information network embedding. In *International Conference on World Wide Web*, 1067–1077.
- Valverde-Rebaza, J., and de Andrade Lopes, A. 2013. Exploiting behaviors of communities of twitter users for link prediction. *Social Network Analysis and Mining* 3(4):1063–1074.
- Ver Steeg, G., and Galstyan, A. 2012. Information transfer in social media. In *International Conference on World Wide Web*, 509–518.
- Ver Steeg, G., and Galstyan, A. 2013. Information-theoretic measures of influence based on content dynamics. In *ACM International Conference on Web Search and Data Mining*, 3–12.
- Wasserman, S., and Faust, K. 1994. *Social network analysis: Methods and applications*, volume 8. Cambridge university press.
- Yang, Y.; Jia, J.; Wu, B.; and Tang, J. 2016. Social role-aware emotion contagion in image social networks. In *AAAI Conference on Artificial Intelligence*.
- Zhang, W.; Wang, J.; and Feng, W. 2013. Combining latent factor model with location features for event-based group recommendation. In *ACM International Conference on Knowledge Discovery and Data Mining*, 910–918.