Modeling the Evolution of Users' Preferences and Social Links in Social Networking Services

Le Wu, Yong Ge, Qi Liu, *Member, IEEE*, Enhong Chen, *Senior Member, IEEE*, Richang Hong, *Member, IEEE*, Junping Du, and Meng Wang, *Member, IEEE*

Abstract—Sociologists have long converged that the evolution of a *S*ocial *N*etworking *S*ervice(SNS) is driven by the interplay between users' preferences (reflected in user-item interaction behavior) and the social network structure (reflected in user-user interaction behavior). Nevertheless, traditional approaches either modeled these two kinds of behaviors in isolation or relied on a static assumption of a SNS. Thus, it is still unclear how do the roles of the dynamic social network structure and users' historical preferences affect the evolution of SNSs. Furthermore, can transforming the underlying social theories in the platform evolution modeling process benefit both behavior prediction tasks? In this paper, we incorporate the underlying social theories to explain and model the evolution of users' two kinds of behaviors in SNSs. Specifically, we present two kinds of representations for users' behaviors: a direct (latent) representation that presumes users' behaviors are represented directly (latently) by their historical behaviors. Under each representation, we associate each user's two kinds of behaviors with two vectors at each time. Then, for each representation, we propose the corresponding learning model to fuse the interplay between users' two kinds of behaviors. Finally, extensive experimental results demonstrate the effectiveness of our proposed models for both user preference prediction and social link suggestion.

Index Terms—User modeling, social networking services, user interest modeling, link prediction

1 Introduction

With the popularity of online social media, many SNSs have emerged in recent years. These SNSs provide an online platform for facilitating the building of social relations among people who share similar consumption interests, activities, or real-life connections. Thus people can stay connected with others and be informed of social friends' consumption preferences. For instance, in the online product review platform *Epinions.com*, the system immediately pushes the product ratings and reviews to a user that she trusts. In a location based social network service website such as *Gowalla*, people share location-embedded information with friends by adding a check-in at that place.

Generally, a SNS platform is built upon two kinds of users' behaviors: *consuming items* (reflected in the *user-item interaction* such as rating, buying and check-in) and *building social links* among users (reflected in the *user-user interaction* such as the directed trust and the undirected friendship). While users face a dazzling array of potential consumption items and unlinked social entities, discovering users' consumption preferences and suggesting new links become two core behavior prediction tasks for these systems. Computational tools have been developed to solve these two tasks respectively. On one hand, *C*ollaborative *F*iltering (CF) forms the basis of user preference discovery, which assumes that users are likely to consume items that are locally popular among the like-minded users with similar consumption history in the past [3], [24], [30]. On the other hand, *N*ode *P*roximity (NP) based models play a central role in social link suggestions, where two users are possibly to form links in the near future if they are structural close in the social graph [28], [32]. In summary, these two kinds of models utilized a particular kind of users' historical behaviors to predict the same kind of behaviors in the future, and usually are well researched in parallel.

Nevertheless, social scientists have long converged that these two kinds of users' behaviors are not isolated. Instead, the interplay between them drives the evolution of SNSs, leading to the dynamic changes of users' preferences and the social network structure over time. There are two social theories that explain this evolution: the *social influence* effect states that users' future preferences are affected by the social network around them, and the *homophily effect* suggests that people tend to associate and bond with others that have similar preferences in the past [4]. Based on these theories, some research works have leveraged one type of users' behavior to help predict another type of behavior [15], [20], [40]. Recently, Yang et al. proposed to jointly model the correlation of these two kinds of users' behaviors in a unified static model, thus achieved better performance than modeling them in an isolated way [45]. However, their static view could not capture the evolution of SNSs.

- *L. Wu, R. Hong, and M. Wang are with the School of Computer and Information, Hefei University of Technology, Hefei, Anhui 230009, China. E-mail: {lewu.ustc, hongrc.hfut, eric.mengwang}@gmail.com.*
- *Y. Ge is with the Department of Management Information Systems, University of Arizona, Tucson, AZ 85721. E-mail: yongge@email.arizona.edu.*
- *Q. Liu and E. Chen are with the School of Computer Science and Technology, University of Science and Technology of China, Hefei, Anhui 230026, China. E-mail: {qiliuql, cheneh}@ustc.edu.cn.*
- *J. Du is with the School of Computer Science, Beijing University of Posts and Telecommunications, Beijing 100088, China. E-mail: junpingd@bupt.edu.cn.*

Manuscript received 29 Mar. 2016; revised 28 Nov. 2016; accepted 18 Jan. 2017. Date of publication 2 Feb. 2017; date of current version 27 Apr. 2017. Recommended for acceptance by C. Domeniconi.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TKDE.2017.2663422

In this paper, we incorporate the underlying social theories for explaining and predicting the evolution of SNSs from an individual perspective, i.e., we try to explain how do the roles of the time-evolving social network structure and users' historical preferences affect each users' future behavior, and predict each user's new consumption preference and social links in the near future. Understanding these questions can not only help the SNS system providers gain more insights into users, but also benefit customers by enjoying more accurate predictive services, including consumption preference recommendation and social link suggestion. However, properly addressing these questions is not an easy task. First, both users' consumption behavior and link behavior mix together to form the evolution of SNSs. As we can only observe the final decisions made by users, how to quantify the contribution of each kind of users' behavior is non-trivial. Furthermore, it is still unclear how to embed the social theories to build connections among users' two kinds of behaviors in the modeling process.

To tackle these challenges, in our previous work [42], we proposed a preliminary latent approach named *E*volving *J*oint *P*rediction (EJP) for modeling the evolution of users' consumption behavior and social link behavior in SNS platforms. Specifically, we associate each user with a latent consumption preference vector at each time. Then we propose a probabilistic approach to fuse the interplay between users' consumption and social behaviors over time, where the user latent consumption preference vector bridges the connection between users' two kinds of behaviors.

This paper further extends our previous work [42]. Specifically, we introduce two kinds of representations to depict users' behaviors in SNSs. The first one is a *direct representation* that presumes each kind of a user's behavior can be directly reflected from her historical behavior, i.e., each user's consumption behavior is depicted in the item space and the social link behavior is represented in the user space. Then each user has a direct consumption vector and a social link vector at each time under the direct representation. This direct representation is simple and intuitive, however, as each user usually has very few consumption (social) records in the huge item (user) space, the extreme data sparsity issue may lead to unsatisfactory performance [3]. Thus, instead of the direct representation, we argue that each user's two kinds of behaviors at each time can be determined by the *latent representation*: a latent consumption vector that shows her consumption preference and a latent social vector that depicts her social link behavior. Then for each representation, we propose a corresponding learning model to fuse the interplay between users' two kinds of behaviors by leveraging the underlying social theories. Finally, extensive experimental results demonstrate the effectiveness of our proposed model for both user preference prediction and social link suggestion. In summary, compared with our preliminary model of EJP [42], the main contributions of this paper are:

- We propose a novel direct model to represent users' two kinds of behaviors in SNS platforms, which has not been explored in our preliminary work. The direct model is intuitive and shows high efficiency.
- In the preliminary EJP model, we represent each user's consumption behavior with latent consumption vector without any latent social link representation [42]. We further advance the previous work by representing users' two kinds of behaviors with both latent vectors, i.e., each user has a latent consumption vector and a latent social link vector at each time. Then we design a more sophisticated model to capture the interplay between users' latent consumption vectors and latent social link vectors, while EJP purely shares the latent consumption vector among users' two kinds of behaviors. Experimental results show the latent model proposed in this paper produces better performance than EJP.

2 RELATED WORK

We summarize the related work into the following three categories.

Collaborative Filtering. Collaborative filtering is a technique to provide personalized item suggestions by discovering users' consumption preferences [3]. Usually, we are given a user-item consumption preference matrix with a few known preference values, and the goal is to predict the unknown values as accurate as possible. Models in this area can be classified into two categories: the neighborhood-based methods and the matrix factorization models. The neighborhood based methods predicted the unknown preference of a target user based on similar neighbor users' ratings for this item [6], [12], [38]. In contrast, matrix factorization models tried to capture both users and items in a low latent space by learning user-item past interactions. After that, a user's unknown preference for an item could be predicted by comparing the correlation of the learned latent vectors between them [24], [34].

In the real world, users' preferences are not static but change over time. Thus, it is important to take the temporal dynamics of users' interests in the recommendation process [8], [19], [23], [35], [44]. E.g., TimeSVD++ was proposed to adapt some of the user latent vectors to evolve over time [23]. Xiong et al. introduced an additional latent time dimension in a factor-based model that captures the population-level preference of products [44]. Jiang et al. proposed a dynamic scheme of tensor factorization for temporal multi-faceted behavior prediction [19]. These models captured users' interest drifts, however, the explanatory reasons for the interest changes are not clear.

Link Prediction. Link prediction is the general problem of predicting the potential new link connections among a social network in a near future. This problem has long been considered from a static view: the input is a snapshot of a social network at time t, and the goal is to predict the possible links that appear from t to a later time t'. Generally, this question can be viewed as computing the node proximity, or similarity given the network topology [28], [32]. The literature on this static link prediction can be classified into two categories: unsupervised and supervised. The unsupervised models used direct neighborhood-based measures (e.g., the percentage of the number of shared friends) or the path-based methods to infer the proximity of indirectly connected users [17], [28]. In contrast to the unsupervised measures, the supervised models took the existing links as positive labels and the current unknown links as possible negative ones, then a learning process is involved with the available data labels. The choice of these supervised models included the feature-based models [26] and the latent-factor based models [32], [33].

In the real world, the social networks are continuously evolving, with new nodes and links added over time. Recent studies considered the temporal link prediction problem, where we have the detailed edge creation time or several snapshots of social networks. An intuitive yet effective approach is to collapse multiple time-sliced linked data into a single matrix with weighted averaging, then the static link prediction models could be applied [1], [11]. Others proposed tensor factorization or non-parametric time-series models to capture the temporal information in graph evolution process [9], [37]. Our work resembles these works in capturing the temporal evolution of the social structural data, nevertheless, our model distinguishes these works in the explanatory ability of the evolution of social networks.

Modeling the Interaction Between Users' Two Kinds of Behaviors. Sociologists have long converged that users' consumption behavior and the social link behavior are not isolated, instead, they have a mutual reinforcement relationship [4], [7]. Specifically, the social influence theory states that people tend to associate and bond with users that have similar preferences and attributes, and the homophily effect suggests that the links between users would further influence users to behave similarly with their friends. Thus, researchers argued that we can leverage one kind of data source for the remaining prediction task. Among them, social-based recommendation system, utilizes the social network structure information to help mitigate the data sparsity issue in traditional recommender systems [14], [15], [18], [20], [27]. Another parallel line is to suggest links by incorporating users' historical consumption preferences. Tang et al. provided an approach to exploit users' historical preferences for more accurate link prediction. They argued that users with similar historical interests are more likely to build social links in the future [40]. More recently, Gong et al. proposed to argument the social network into a social-attribute network, such the node attribute and social network information could inform each other [13]. However, all these models relied on a static assumption of a SNS platform. Jamali analyzed and modeled the temporal behaviors of users in a SNS using bidirectional effects of rating patterns and social relations [16]. This line of work differs from our problem formulation in that it focused on the global evolution of a SNS. To the best of our knowledge, Yang et al. made one of the first few attempts that mutually modeled users' preferences and the social link prediction in a unified framework [45]. The key idea of their proposed model is to define a shared static user latent vector over these two kinds of tasks, thus achieved better performance. Our proposed models advances their technique in several aspects: (1) We made a comprehensive study of users' behaviors with both direct representations and latent representations. (2) In the proposed latent representation, their model assumed a static representation of a SNS while our work captures the temporal dynamics.

3 Problem Definition

In a SNS platform, there are a set of users U ($|U|=N$) and a set of items V ($|V|=M$). Users perform two kinds of behaviors in most SNS platforms: *consuming items* and *building social links* with others. Generally speaking, consuming an item refers to the interaction between a user and an item, which varies at different SNS platforms. Specifically, we represent users' two kinds of behaviors at each time t as two matrices: a consumption matrix $C^t \in \mathbb{R}^{N\times M}$ and a social link matrix $S^t \in \mathbb{R}^{N\times N}$. If user a consumes item i at t, C^t_{ai} denotes the rating preference score. Otherwise it is 0 indicating the user does not show any preference during that time. Similarly, $S^t_{ab}=1$ if user a connects to user b at time t, otherwise it equals 0. As we consider the temporal evolution of SNSs, we summarize users' two kinds of behaviors over time as two matrix sequences: a consumption matrix sequence $C=[C^1,\ldots C^t \ldots, C^T]$ and a social link matrix sequence $S=[S^1,\ldots S^t \ldots, S^T]$. Without confusion, we use a, b, c to represent users and i, j, k to denote items. Then the problem can be defined as:

Definition 1 (Problem Definition). *Given the user consumption sequence C and the social link sequence S, our goal is two-fold: (1) quantify the social influence and homophily effect of each user for the evolution process of SNSs. (2) predict each user's consumption behavior and the social link behavior at time $T+1$.*

4 Direct Modeling of the Evolution of SNS

In this section, we introduce a simple direct approach to model the evolution of SNSs. The key idea of this direct model is that, each kind of a user's future behavior is influenced directly by others' historical behaviors. With this direct representation, uses' consumption behavior (social behavior) over time can be described as vectors in the item space (user space). Then we model the interplay of users' two kinds of behaviors with the direct representation. In fact, this direct approach is quite intuitive and has been integrated into many SNSs. E.g., the social influence explanation of "your friends A and B also like this product" for consumption recommendation process, and the homophily effect explanation of " user B also consumed item a and b" for suggesting social links. Fig. 1 illustrates an example of users' behaviors over time in SNS platforms, with the main reasons for users' new behaviors listed in the right of this figure. Formally, we define the direct modeling as:

Definition 2 (Direct Modeling). *Given the user consumption matrix sequence C and the social sequence S, the direct modeling represent each user's consumption behavior in the item space $V\in\mathbb{R}^N$ and social behavior in the user space $U\in\mathbb{R}^M$. The goal of direct modeling is to solve the problem in Definition 1 based on the direct representation.*

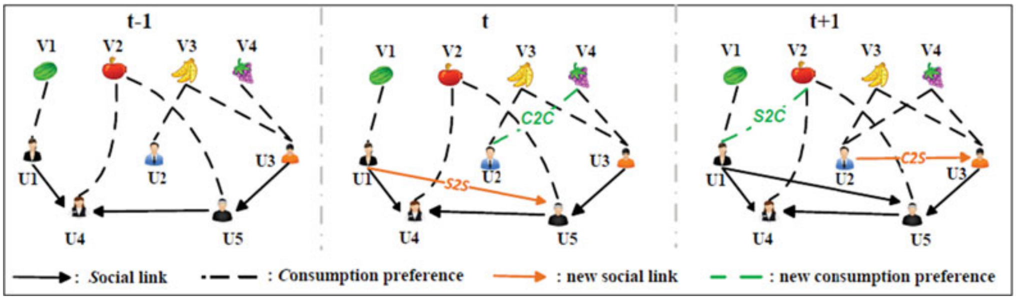
In the following, we first introduce how to capture users' behaviors with the direct representation, then give the corresponding learning model.

4.1 The Proposed Direct Model

Evolutional Direct Consumption Behavior Modeling. As shown in Fig. 1, there are two main reasons for users to build a new consumption record. First, traditional CF models assume a user likes to consume products that are locally popular among users that have similar historical consumption preference. E.g., a possible reason for $u2$ to consume $v4$ is that $u3$—a user that has similar consumption preference with her consumed $v4$. Second, the social influence theory suggests that users are also likely to be influenced by social network neighbors' preferences to make consumption preference decisions. E.g., $u1$ is influenced by her social friend $u4$ and consumes $v2$ at time $t+1$. In summary, for each user's consumption behavior from time window $t=2,3,\ldots,T$, we model it as a liner combination of the collaborative filtering and the social influence effect

$$\hat{C^t_{ai}} = (1-\alpha_a)f(a,i,t) + \alpha_a g(a,i,t) \quad s.t. \quad 0 \le \alpha_a \le 1. \quad (1)$$

where C^t_{ai} denotes the predicted consumption of user a to item i. Function f and g capture the collaborative filtering and the social influence for consumption prediction respectively,



Explanations of A2B

A2B	Explanations
C2C	Collaborative Filtering
S2C	Social Influence
S2S	Node Proximity
C2S	Homophily Effect

Fig. 1. A showcase of the evolution of a SNS platform. At each time, users perform two behaviors: build a new *S*ocial link or show her new *C*onsumption preference. For simplicity, we use "A2B" at each new added behavior to denote the current behavior of "A" leads to the future behavior "B". For example, a "S2C" label is added from t to $t+1$ as $u1$ shows *C*onsumption preference to $v2$ possibly because her social neighbors ($u1$ has followed $u4$ and $u5$ before $t+1$) consumed item $v2$ before.

which are balanced by the non-negative parameter α_a for each user. As users may have their own decisions in balancing these two aspects, e.g., some users like to follow their own historical preferences to make future consumptions (i.e., small α_a) while others prefer to receive consumption suggestions from their friends (i.e., large α_a), the balance parameters are personalized and vary from person to person. Specifically, function f captures the predicted consumption score by utilizing users' consumption history, thus any CF models, such as the item-based collaborative filtering [38], latent factor models [34] could be applied. As the focus of this paper is not to devise more sophisticated models to predict the consumption preference by traditional CF models, we use any available CF algorithm. Next we introduce how to build social influence effect function g.

$g(a,i,t)$ is a function that models the *social influence* effect for user a's consumption preference on product i at time t. The social influence effect states the diffusion of information over social networks that lead people to consume local popular items among their social friends [10]. This effect has been viewed as a foundation for many important social applications, such as the social influence maximization for viral marketing [21], [43]. With this direct social influence effect, each user a's future preference on product i is influenced directly by the historical consumption records of her social neighbors' decisions of the same item

$$g(a,i,t) = \frac{\sum_{t'=1}^{t-1}\sum_{b\in N_a^{t'}} Y_{bi}^{t'} s_{ab}^{t} C_{ab}^{t'}}{\sum_{t'=1}^{t-1}\sum_{b\in N_a^{t'}} Y_{bi}^{t'} s_{ab}^{t}}, \tag{2}$$

where N_a^t is the set of users that a connects till t. Y_{bi}^t is an indicator value that equals 1 if b consumes i at time t, otherwise it equals 0. In this equation, s_{ab}^t denotes the social influence strength between this pair of users, i.e., the social proximity of these two users based on the social network structure. As the social network is dynamic and changes over time, it is reasonable to assume the social influence strength also varies. There are a number of ways to calculate the social influence strength. For example, we can adapt the traditional static node proximity based models to a dynamic version or any dynamic link proximity based model. Here, we simply adopt a widely used node proximity measure—Adamic/Adar metric [28], and adapt it to a time varying version as

$$S_{ab}^t = \frac{1}{\sum_{c\in N_a^{t-1}\cap N_b^{t-1}} log(|N_c^{t-1}|)}. \tag{3}$$

Evolutional Direct Link Behavior Modeling. Similar as the direct consumption preference prediction in Eq. (1), a user's decision on whether to build a social link is also mainly determined by two factors. First, two users are likely to form links in the future if they are topologically close to each other from the social network structure. This node-proximity phenomenon forms the basis of traditional link prediction algorithms. E.g., $u1$ bonds with $u5$ at time t in Fig. 1 can be mainly attributed to this reason. Second, the preference similarity between users also brings connections, i.e., user $u2$ finds $u5$ has many common consumption preferences with her (they both consumed $v4$ and $v5$), then $u2$ is likely to associate with $u5$ in the near future as shown in Fig. 1. This is termed as the homophily effect and it is widely accepted by social scientists in explaining the social network construction process [31]. Based on these two effects, each user a's predicted link score to user b, denoted as $\hat{S_{ab}^t}$ at time t ($t=2,\ldots,T$), is modeled as

$$\hat{S_{ab}^t} = (1-\beta_a)h(a,b,t) + \beta_a l(i,j,t) \quad s.t. \quad 0 \le \beta_a \le 1. \tag{4}$$

where function h and l calculates the node proximity and the homophily effect for link prediction. β_a is a personalized parameter that balances these two effects. Since we do not focus on modeling node proximity in this paper, any traditional node proximity models could be used for calculating h. In the following, we introduce how to model the homophily effect function l under the direct assumption.

Basically, $l(a,b,t)$ is a function that explains the *homophily effect* for social link connection among user pair a and b at time t, sometimes which is also termed as the social selection process. This effect states that the tendency of users to form social connections with similar characteristics and preferences, e.g., we like to form friendships with others that have similar backgrounds, education level and lots of interests in common. This pervasive phenomenon has a long history of study in sociology and happens in our everyday life [25], [31]. Given the direct representation of users' behaviors, the homophily effect between two users can be directly obtained by comparing the consumption records between them. Here, we simply adapt the cosine similarity measure to calculate the homophily effect between any pair of users. Let $L^t(a) \in \mathbb{R}^{M\times 1}$ denote the vector of user a's consumption records over the item space before time t, i.e., the j's element in this vector equals 0 if user a did not consume item j before t, otherwise it equals $C_{aj}^{t'}$ if $\exists t' < t$, $Y_{aj}^{t'} = 1$. Then the homophily effect between a and b at time t is

$$l(a,b,t) = cos(L^t(a), L^t(a)) = \frac{< L^t(a), L^t(b) >}{||L^t(a)||_F * ||L^t(b)||_F}, \tag{5}$$

where $<,>$ denotes the inner product of two vectors and $||L^t(a)||_F$ is the Frobenius form of this vector.

4.2 Model Learning

After defining the direct social influence effect function g and the homophily effect function l, in order to predict users' future behaviors as shown in Eqs. (1) and (4), we need to determine the balance parameters $\alpha = [\alpha_a]_{a=1}^N$ and $\beta = [\beta_a]_{a=1}^N$ for all users. Specifically, with the availability of users' historical consumption behavior in SNSs, we define the loss function as

$$\begin{aligned}\min_{\alpha} \mathcal{L}(\alpha) =&\sum_{t=2}^{T}\sum_{a=1}^{N}\sum_{i=1}^{M} Y_{ai}^t (C_{ai} - \hat{C}_{ai})^2 \\ =&\sum_{t=2}^{T}\sum_{a=1}^{N}\sum_{i=1}^{M} Y_{ai}^t (C_{ai} - (1-\alpha) f(a,i,t) - \alpha_a g(a,i,t))^2 \\ s.t.&\quad \forall a \in U, \quad 0 \le \alpha_a \le 1.\end{aligned} \tag{6}$$

The above loss function is convex with bound constraints, thus a global minimum could be achieved. In practice, we could resort to Projected Gradient Descent (PGD) method [29]. Specifically, for each α_a ($0 \le \alpha_a \le 1$), the PGD method updates the current solution α_a^k in kth iteration to α_a^{k+1} according to the following rule

$$\alpha_a^{k+1} = P[\alpha_a^k - \eta \nabla_{\alpha_a}],\ P(\alpha_a) = \begin{cases} \alpha_a & \text{if} \quad 0 \le \alpha_a \le 1, \\ 0 & \text{if} \quad \alpha_a < 0, \\ 1 & \text{if} \quad \alpha_a > 1, \end{cases} \tag{7}$$

where ∇_{α_a} has the following form

$$\nabla_{\alpha_a} = \sum_{t=2}^{T}\sum_{i=1}^{M} Y_{ai}^t (\hat{C}_{ai}^t - C_{ai}^t) \times (g(a,i,t) - f(a,i,t)).$$

We summarize the framework of the proposed direct consumption behavior modeling process in Algorithm 1.

Algorithm 1. The Direct Approach for Consumption Behavior Prediction

Input: A traditional CF function f;
Output: The predicted consumption preference at $T+1$;
Initialize α with small positive values;
for all (a,i,t) in consumption training dataset **do**
  calculate $f(a,i,t)$ based on the given CF algorithm;
  calculate $g(a,i,t)$ (Eq. (2));
  predict consumption preference (Eq.(1);
end for
while the loss function of Eq. (6) does not converge **do**
  for user $a = 1;\ a \le N;\ a++$ **do**
    update α_a using PGD (Eq. (8));
  end for
end while
for all $(a,i,T+1)$ records in consumption test dataset **do**
  calculate $f(a,i,T+1)$ based on the given CF algorithm;
  calculate $g(a,i,T+1)$ (Eq. (2));
  predict consumption preference based on Eq. (1);
end for

Similarly, for the balance vector β in the social link prediction task, we can also formulate a loss function as

$$\begin{aligned}\min_{\beta} \mathcal{L}(\beta) &= \sum_{t=2}^{T}\sum_{a=1}^{N}\sum_{b=1}^{N} Z_{ab}^t (S_{ab}^t - \hat{S}_{ab})^2 \\ &= \sum_{t=2}^{T}\sum_{a=1}^{N}\sum_{i=1}^{M} (S_{ab} - (1-\beta_a) h(a,b,t) - \beta_a l(a,b,t))^2.\end{aligned} \tag{8}$$

Then we could also resort to PGD method to update β_a.

Dealing with Data Imbalance. Note that in social link construction process, $S_{ab}^t = 0$ denotes a missing link between user a and b. If we consider all missing link records in the optimization function of Eq. (8), the problem turns to a highly imbalanced learning problem with much more labels of 0 than 1. Here, we borrow an effective undersampling technique. Particularly, for each newly added positive link, we randomly select m missing links as observed pseudo negative links with a weight of $\frac{1}{m}$ at each iteration in parameter learning process. Since the sampling process is random and each time the negative samples change, each missing link gives very weak negative signal [16], [32].

4.3 Time Complexity

The direct approach is simple and intuitive. In fact, as most SNS platforms have already implemented the traditional collaborative filtering and node proximity models (i.e., function f and h), the time complexity for this model mainly lies in two parts as shown in Algorithm 1: first calculating the predicted direct social influence score g and homophily score l, and then learning the balance parameters. Suppose there are c non-empty consumption records in consumption matrix sequence C and s social links in social matrix sequence S ($c \ll M \times N, s \ll N \times N$), then the average consumption records and social connections of each user are $t_c = \frac{c}{N}$ and $t_s = \frac{s}{N}$, which are usually very small in practice. Specifically, the time complexity is $O(c \times t_s)$ for calculating function g and $O(s \times t_c)$ for function l. In parameter learning process, since we only have two parameters for each user (i.e., α_a and β_a), the time complexity is $O(N \times (t_c + t_s)) = O(c+s)$ for each iteration. Thus the total time complexity of the direct approach is $O(c \times t_s + s \times t_c)$.

5 Latent Modeling of the Evolution of SNS

We have already introduced how to model and predict users' behaviors under the direct representation, where each user's two kinds of behaviors are directly represented in the item space and user space. In the real world SNSs, the user space and item space are very huge, with hundreds of thousands of users and items. Given limited social links and consumption records of each user, the direct approach may suffer from the cold-start problem: it could hardly calculate reliable direct social influence strength (Eq. (3)) and homphily effect (Eq. (2)) score if a pair of users share very few common consumption (social link) records [3]. For example, *Alice* liked romantic movies such as *Roman Holiday* and *Titanic*, and *Bob* evaluated highly for movies *Sleepless in Seattle* and *Notting Hill*. Since they have common latent consumption preferences for romantic movies, based on the homophily theory, they are probably like to build social connections in the near future. Nevertheless, the above direct approach can hardly capture the homophily effect between them as they did not consume a common item. Thus, we argue that the interplay between users' behaviors may be latent, i.e., each user's consumption preference and her social link structure could be depicted by a small number of latent dimensions. In this section, we propose a latent approach to model the evolution of users' behaviors. We define the latent modeling approach as follows:

Definition 3 (Latent Modeling). *Given the user consumption matrix sequence C and the social sequence S, the latent modeling approach aims to learn a function to map users' consumption and social behavior into a low-dimensional latent space*

$\mathbb{R}^D, D \ll M, N$. Each user u's consumption behavior and social behavior at each time t can be represented as a latent consumption vector $U^t \in \mathbb{R}^D$ and a latent social vector $P^t \in \mathbb{R}^D$ in the latent space. The goal of latent modeling is to design a model to learn the latent consumption vectors and the latent social vectors of users, and the solve the problem in Definition 1 based on the latent representation.

5.1 The Proposed Latent Model

Here,we introduce how to model the evolution of each kind of a user's behaviors with the latent representation.

Evolutional Latent Consumption Behavior Modeling. With the latent representation, for each user a and each item i, the predicted consumption preference between them at time t could be expressed as

$$p(C|U,V) = \prod_{t=1}^{T}\prod_{a=1}^{N}\prod_{i=1}^{M}\mathcal{N}[(C_{ai}^t| < U_a^t, V_i >, \sigma_C^2)]^{Y_{ai}^t}, \quad (9)$$

where $\mathcal{N}(\mu, \sigma^2)$ is a normal distribution with mean μ and variance σ^2. Y_{ai}^t is an indicator variable that equals 1 if user a consumes item i at time t, otherwise it equals 0. $U_a^t \in \mathbb{R}^{D\times 1}$ is the latent consumption vector of a at time t in user latent matrix U^t and $V_i \in \mathbb{R}^{D\times 1}$ is the item latent vector in item latent matrix $V \in \mathbb{R}^{M\times D}$. To model users' preference changes, we assume users' latent matrix vary among time in the above equation. Thus we can summarize users' latent preferences into a latent consumption preference matrix sequence $U = [U^1, \ldots, U^t, \ldots, U^T]$. Given the limited observed preference data, a typical approach to avoid overfitting is to add priors to the latent variables. As traditional matrix factorization models [34], we add zero-mean Gaussian priors on the item latent matrix

$$p(V|\sigma_V^2) = \prod_{i=1}^{M}\mathcal{N}(V_i|0, \sigma_V^2\mathbf{I}). \quad (10)$$

Now our goal turns to model the evolution of users' latent consumption matrix sequence U. As illustrated before, for each user, both her previous latent consumption preference and the social influence effect influence a user's future latent interest. Since we use the latent representation of users' consumption interests, we explicitly model the two effects of each user's latent interests at time window $t = 2, 3, \ldots T$ as

$$\begin{aligned} & p(U_a^t) = \mathcal{N}(U_a^t|\bar{U}_a^t, \sigma_U^2\mathbf{I}) \\ \text{where} \quad & \bar{U}_a^t = (1-\alpha_a)U_a^{(t-1)} + \alpha_a \sum_{b\in N_a^{(t-1)}} \frac{s_{ab}^{t-1}}{F_a^{t-1}} U_b^{(t-1)} \\ s.t. \quad & \forall a \in U, \quad 0 \le \alpha_a \le 1, \end{aligned} \quad (11)$$

where s_{ab}^{t-1} denotes the influence of user b to a at time $t-1$. $F_a^{t-1} = \sum_{b\in N_a^{(t-1)}} s_{ab}^{t-1}$ is a normalization constant over all of a's friends at time $t-1$. This normalization ensures that $\sum_{b\in N_a^{(t-1)}} \frac{s_{ab}^{t-1}}{F_a^{t-1}} = 1$. The social influence score s_{ab}^{t-1} depicts how similar these two users are in the social space. Since we also characterize each user with a latent structure factor P_a^{t-1}, it is nature to enforce users that have similar latent structure factor with a get larger influence strength scores. Therefore, we define the influence strength score as

$$S_{ab}^{t-1} = e(< P_a^{t-1}, P_b^{t-1} >) = \frac{1}{1+exp(- < P_a^{t-1}, P_b^{t-1} >)}, \quad (12)$$

where $e(x) = \frac{1}{1+exp(-x)}$ is a logistic function that bounds the influence score in range $(0, 1)$.

At the initial time $t=1$, the social network has not been set up yet, thus each user's latent consumption preferences are only determined by her own consumption preferences without any social influence. We assume a zero-mean Gaussian distribution of users' latent vectors at that time. Then we summarize the prior over user latent consumption matrix sequece as

$$p(U|\sigma_U^2, \sigma_{U1}^2) = \prod_{a=1}^{U}\mathcal{N}(U_a^1|0, \sigma_{U1}^2\mathbf{I})\prod_{t=2}^{T}\mathcal{N}(U_a^t|\overline{U_a^t}, \sigma_U^2\mathbf{I}). \quad (13)$$

Evolutional Latent User Social Behavior Modeling. Each user a's link behavior at time t ($t = 2, 3, \ldots T$) is also mainly influenced by two factors: the node proximity in the social graph and the homophily effect between users. For each user a, as we have her latent social link vector P_a^t and the latent consumption vector U_a^t, we model the link score at time $t = 2, 3, \ldots, T$ as

$$\begin{aligned} & \hat{S}_{ab}^t = (1-\beta_a) < P_a^{t-1}, P_b^{t-1} > + \beta_a < U_a^{(t-1)}, U_b^{(t-1)} >, \\ & s.t. \quad \forall a \in U, \quad 0 \le \beta_a \le 1, \end{aligned} \quad (14)$$

where $\hat{S}_{ab}^t$ is the predicted link score between a and b at time t. In this equation, the link based similarity is computed as the closeness of their latent structure vectors: P_a^t and P_b^t. Similarly, the homophily effect between them is measured by the closeness of their latent consumption vectors.

At time slice $t=1$, no historical user latent consumption preference is available. We assume the social structure proximity is the only effect that determines the social relationships

$$\hat{S}_{ab}^1 = < P_a^1, P_b^1 > . \quad (15)$$

Given the predicted link score in Eq. (14), the likelihood of the predicted link value could be modeled as

$$p(S|U,P,\sigma_S) = \prod_{t=2}^{T}\prod_{a=1}^{N}\prod_{b=1}^{N}\mathcal{N}[(S_{ab}^t|\hat{S}_{ab}^t, \sigma_S^2)]^{Z_{ab}^t}, \quad (16)$$

where Z_{ab}^t denotes whether a builds a link to b at time t.

For each user a, a reasonable assumption is that her latent structure vector P_a^t varies smoothly over time

$$p(P_a^t) = \mathcal{N}(P_a^t|P_a^{t-1}, \sigma_U). \quad (17)$$

At the initial time slice, no previous users' consumption is available and we assume

$$p(P_a^1) = \mathcal{N}(P_a^1|0, \sigma_{U1}). \quad (18)$$

By combining Eqs. (17) and (18), the latent social matrix sequence P has a prior

$$p(P|\sigma_U^2, \sigma_{U1}^2) = \prod_{a=1}^{N}\mathcal{N}(P_a^1|0, \sigma_{U1}^2\mathbf{I})\prod_{t=2}^{T}\mathcal{N}(P_a^t|P_a^{t-1}, \sigma_U^2\mathbf{I}). \quad (19)$$

5.2 Model Learning

Given the users' behavior sequences C and S, our goal is to learn the parameter set $\Phi = [U, P, V, \alpha, \beta]$, where $\alpha = [\alpha_a]_{a=1}^N$, $\beta = [\beta_a]_{a=1}^N$. Specifically, we have the posterior distribution over the parameters Φ as

$$p(U,P,V,\alpha,\beta|C,S) \propto p(C|U,P,V,\alpha)p(S|U,P,\beta)p(U)p(P)p(V). \quad (20)$$

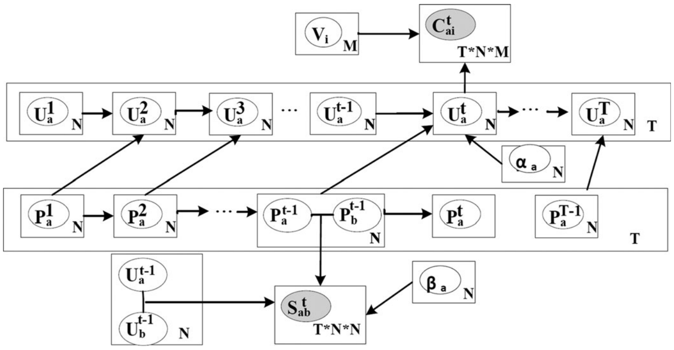


Fig. 2. The graphical representation of the proposed latent model.

We summarize the graphical representation of the proposed latent model in Fig. 2, where the shaded and unshaded variables indicate the observed and latent variables respectively. Given this graphical model, maximizing the log posterior of the Eq. (20) is equivalent to minimizing the following objective

$$
\begin{aligned}
\min_{\Phi} \mathcal{E}(\Phi) &= \frac{1}{2}\sum_{t=1}^{T}\sum_{a=1}^{N}\left\{\sum_{i=1}^{M} Y_{ai}^{t}[\hat{C}_{ai}^{t} - C_{ai}^{t}]^2 + \lambda_S \sum_{b=1}^{N} Z_{ab}^{t}[\hat{S}_{ab}^{t} - S_{ab}^{t}]^2\right\}\\
&+\frac{1}{2}\lambda_U \sum_{t=2}^{T}\sum_{a=1}^{N}(||\overline{U_a^t} - U_a^t||_F^2 + ||P_a^t - P_a^{t-1}||_F^2)\\
&+\frac{1}{2}\lambda_{U1}\sum_{a=1}^{N}(||U_a^1||_F^2 + ||P_a^1||_F^2) + \frac{1}{2}\lambda_V \sum_{i=1}^{M}||V_i||_F^2\\
s.t. \quad & \forall a \in U, 0 \le \alpha_a \le 1, 0 \le \beta_a \le 1,
\end{aligned} \tag{21}
$$

where $\lambda_S = \frac{\sigma_C^2}{\sigma_S^2}$, $\lambda_U = \frac{\sigma_C^2}{\sigma_U^2}$, $\lambda_{U1} = \frac{\sigma_C^2}{\sigma_{U1}^2}$ and $\lambda_V = \frac{\sigma_C^2}{\sigma_V^2}$.

The coupling between U, P, V and the balance parameters makes the above loss function not-convex. In practice, a local minimum could be achieved by performing gradient descent on each parameter iteratively. Specifically, for each user a and each item i, the derivative of each parameter is

$$
\begin{aligned}
\nabla_{V_i} &= \sum_{t=1}^{T}\sum_{a=1}^{N} Y_{ai}^{t}(\hat{C}_{ai}^{t} - C_{ai}^{t})U_a^t + \lambda_V V_i\\
\nabla_{\alpha_a} &= \lambda_U \sum_{t=2}^{T}(\overline{U_a^t} - U_a^t)\left\{\sum_{b\in N_a^{(t-1)}} \frac{s_{ab}^{t-1}}{F_a^{t-1}} U_b^{(t-1)} - U_a^{(t-1)}\right\}\\
\nabla_{\beta_a} &= \lambda_S \sum_{t=2}^{T}\sum_{b=1}^{M} Z_{ab}^{t}(\hat{S}_{ab}^{t} - S_{ab}^{t})\times\\
&\qquad (< P_a^{(t-1)}, P_b^{(t-1)} > - < U_a^{(t-1)}, U_b^{(t-1)} >).
\end{aligned} \tag{22}
$$

The gradient of user latent consumption vector U_a^t and user latent structure vector P_a^t are

$$
\begin{aligned}
\nabla_{U_a^t} &= \sum_{i=1}^{M} Y_{ai}^{t}(\hat{R}_{ai}^{t} - R_{ai}^{t})V_j + \mathcal{I}[t=1]\lambda_{U1}U_a^1 + \mathcal{I}[t\ge 2]\lambda_U(U_a^t - \overline{U_a^t})\\
&+ \lambda_U(1-\alpha_a)(\overline{U_a^{(t+1)}} - U_a^{(t+1)}) + \lambda_U \sum_{a\in N_c^t} \alpha_c \frac{s_{ca}^t}{F_c^t}(\overline{U_c^{(t+1)}} - U_c^{(t+1)})\\
&+ \mathcal{I}[t < T]\lambda_S \beta_a \sum_{b=1}^{N} Z_{ab}^{(t+1)}(\hat{S}_{ab}^{(t+1)} - S_{ab}^{(t+1)})U_b^t\\
&+ \mathcal{I}[t < T]\lambda_S \sum_{c=1}^{N} Z_{ca}^{(t+1)}(\hat{S}_{ca}^{(t+1)} - S_{ca}^{(t+1)})(\beta_c U_c^t),
\end{aligned} \tag{23}
$$

$$
\begin{aligned}
\nabla_{P_a^t} &= \mathcal{I}[t=1]\lambda_S\left[\sum_{b=1}^{N} Z_{ab}^{t}(\hat{S}_{ab}^{t} - S_{ab}^{t})P_b^t + \sum_{c=1}^{N} Z_{ca}^{t}(\hat{S}_{ca}^{t} - S_{ca}^{t})P_c^t\right]\\
&+ \lambda_S \sum_{b=1}^{N} Z_{ab}^{t+1}(\hat{S}_{ab}^{t+1} - S_{ab}^{t+1})(1-\beta_a)P_b^t\\
&+ \lambda_S \sum_{c=1}^{N} Z_{ca}^{t+1}(\hat{S}_{ca}^{t+1} - S_{ca}^{t+1})(1-\beta_c)P_c^t\\
&+ \lambda_U \alpha_a \sum_{b\in N_a^t}\left\{\frac{F_a^t \frac{\partial s_{ab}^t}{\partial P_a^t}\times U_b^{\prime t} - s_{ab}^t(\sum_{l\in N_a^t}\frac{\partial s_{al}^t}{\partial P_a^t})\times U_b^{\prime t}}{(F_a^t)^2}\right\}(\overline{U_a^{t+1}} - U_a^{t+1})\\
&+ \lambda_U \sum_{a\in N_c^t}\frac{\alpha_c\{(F_c^t - s_{ca}^t)\frac{\partial s_{ac}^t}{\partial P_a^t}U_a^{\prime t}\}}{(F_c^t)^2}(\overline{U_c^{t+1}} - U_c^{t+1})\\
&+ \lambda_{U1}\{\lambda_{U1}P_t^1 + \lambda_a(P_t^1 - P_t^2)\} + \mathcal{I}[t = T-1]\lambda_a(P_a^{T-1} - P_a^{T-2})\\
&+ \mathcal{I}[1 \le t < T-1]\lambda_U(2P_a^t - P_a^{t+1} - P_a^{t-1}).
\end{aligned} \tag{24}
$$

where $\mathcal{I}[x]$ is an indicator function that equals 1 if x is true and 0 otherwise. The derivative of the $\frac{\partial s_{ab}^t}{P_a^t}$ is

$$
\frac{\partial s_{ab}^t}{\partial P_a^t} = s_{ab}^t * (1 - s_{ab}^t)P_b^t. \tag{25}
$$

For the updating step, as there are no constraints on U, P and V, we can update them directly using Stochastic Gradient Descent (SGD) method [5]. With the bound constraints of α_a and β_a, a local minimum can be found by the Projected Gradient Descent method.

5.3 Prediction

After learning the related parameters $\Phi = [U, V, \alpha, \beta]$, the two goals in the problem definition process can be answered: (1) the relative contribution of the social influence and the homophiy effect for the evolution process of each user's future consumption and link behaviors can be directly obtained from parameters α and β. (2) The predicted behaviors of each user at $T+1$ are

$$
\begin{aligned}
U_a^{(T+1)} &\approx (1-\alpha_a) * U_a^T + \alpha_a \sum_{b\in N_a^T} s_{ab}^{T-1}U_b^T.\\
\hat{C}_{ai}^{(T+1)} &= U_a^{(T+1)}\cdot V_i \approx (1-\alpha_a) * U_a^T + \alpha_a \sum_{b\in N_a^T}\frac{s_{ab}^T}{F_a^T}U_b^T \cdot V_j,\\
\hat{S}_{ab}^{(T+1)} &= (1-\beta_a) < P_a^T, P_b^T > + \beta_a U_a^T \cdot U_b^T.
\end{aligned} \tag{26}
$$

5.4 Time Complexity

We summarize the latent approach model in Algorithm 2. The time complexity of the latent approach mainly lies in computing the user latent consumption sequence U, the user social link sequence P, the item latent matrix V and the balance parameters. Specifically, in each iteration, the time complexity is $O(N\times T\times D\times(\frac{t_c}{T}+\frac{t_s}{T}+t_s\times D) = O(D\times(c+T\times D\times s))$, for both U and P, $O(D\times c)$ for V, and $O(c+s)$ for the balance parameters. Thus the total complexity is $O(D\times(c+T\times D\times s))$, which is linear with the records and the time windows.

6 Experiments

In this section, we conduct experiments on two real-world datasets. Specifically, we demonstrate: (1) the effectiveness of our proposed models (Sections 6.2 and 6.3); (2) the efficiency comparison (Section 6.4); (3) understanding on the balance parameters (Section 6.5); (4) the setting of several parameters in our proposed models (Section 6.6).

Algorithm 2. The Latent Approach for Behavior Prediction

Input: The consumption sequence C and social link sequence S
Output: The predicted consumption and link preference at time $T+1$
Initialize U, V, P, α and β
while the loss function of Eq. (21) does not converge **do**
  for user $a=1;\ \ a \leq N;\ \ a++$ **do**
    for time $t=1;\ \ t \leq T;\ \ t+=$ **do**
      Fix P, V, α, β, update U_a^t using SGD ;
      Fix U, V, α, β, update P_a^t using SGD;
    end for
    Fix U, P, V, update α_a and β_a using PGD;
  end for
  for product $i=1;\ \ i \leq M;\ \ i++$ **do**
    Fix U, P, α, β, update V_i using SGD;
  end for
end while
for all $(a, i, T+1)$ records in consumption test dataset **do**
  predict consumption preference based on Eq. (26);
end for
for all $(a, b, T+1)$ records in link test dataset **do**
  predict potential social link score based on Eq. (26);
end for

6.1 Data Description and Experimental Setup

The datasets we used are: the who-trust-whom online product sharing dataset *Epinions* [36] and the location based social networking dataset *Gowalla* [39]. In both datasets, we treated each month as a time window. We filtered out users that have less than 2 consumption records and 2 social links. After that, each user's preference rating is normalized into 0 to 1. Table 1 shows the basic statistics of the two datasets after pruning. Specifically, in data splitting process, we use the data till time T for model training, i.e., T=11 (T=3) in Epinion (Gowalla). Among them, we randomly extract 10 percent of the records as validation data for parameter tuning. The newly added behaviors in T+1 are treated as the test data.

In the following, we report the results of our proposed two models. Particularly, we call the direct approach introduced in Section 4 as *E*volving *D*irect *C*onsumption *P*rediction (EDCP) and *E*volving *D*irect *L*ink *P*rediction (EDLP) for users' two kinds of behaviors respectively, and the model proposed in Section 5 with latent representation of users' behaviors as *E*volutional *L*atent *J*oint *P*rediction (ELJP) that jointly models users' two kinds of behaviors. To further validate the effectiveness of jointly modeling users' two kinds of behaviors with ELJP, we have also devised two simplified models of ELJP: *E*volving *L*atent *C*onsumption *Prediction* (ELCP) and *E*volving *L*atent *L*ink *Prediction* (ELLP). Specifically, ELCP leverages the dynamic social network for consumption prediction (i.e.,$\lambda_S=0$ in Eq. (21)) and ELLP utilizes users' temporal consumption preferences for link prediction (i.e., we do not optimize the first term in Eq. (21)). There are several parameters for these models. The latent dimension is set as $D=10$ in ELJP. We set the regularization parameters as $\lambda_S=0.3, \lambda_{U1}=\lambda_V=0.1$. The user regularization parameter λ_U is set to be 5 in Epinions and 1 in Gowalla. Also, we call the model proposed in our previous work [42] as *E*volving *J*oint *P*rediction for consistency. Users could refer to our previous work for more details of EJP [42].

TABLE 1
The Statistics of the Two Datasets

Dataset	Epinions	Gowalla
Users	4,630	21,755
Items	26,991	71,139
Time Windows	12	4
Training Consumptions	62,872	278,154
Training Links	75,099	251,296
Test Consumptions	2,811	52,448
Test Links	3,257	6,254
Consumption Density	0.050%	0.018%
Link Density	0.35%	0.053%

6.2 User Consumption Preference Prediction

We compare consumption prediction results with:

- PMF: PMF tried to project users and items into the same low latent space by mining the sparse consumption matrix [34].
- TMF: This model extended static collaborative filtering models by capturing the evolving nature of users' preferences over time [44].
- SocialMF: This model belongs to social-based recommender system. Specifically, the authors incorporated the social influence effect for users' preference modeling [15].
- ContextMF: ContextMF utilized the social contextual information and summarized the knowledge as the individual preference and interpersonal influence for recommendation [20].
- SAN: This method augmented the social network structure and node attributes into a social-attribute network to perform both link prediction and attribute inference. Particularly, we treat the items that each user consumes as the attributes of this user [13].
- FIP: This method jointly modeled users' preference and social link prediction in a unified framework [45]. Specifically, FIP defined a shared static latent factor that showed both a user's consumption preference and the link behavior [45].

For better illustration, we summarize the details of these models in Table 2. We adopt the widely used *R*oot *M*ean *S*quared *E*rror (RMSE) as the evaluation metric for rating prediction precision comparison [34], [44]. There are several parameters in these baselines, for fair comparison, we carefully tuned these parameters in the validation set to ensure the best performance. In our proposed direct consumption prediction model EDCP, we need to determine the f function that adopts traditional collaborative filtering technique to predict users' future consumption behavior. As both the baseline of PMF and TMF are valid for the f function, we choose the best result among these two baselines.

Fig. 3 shows the experimental results of various models on the test data with the varying parameter of the latent dimension size D on both datasets. There are several observations. First, TMF, SocialMF and ContextMF perform better than PMF, indicating the effectiveness of incorporating the time and social network information for user preference

TABLE 2
Characteristics of the Baselines, with C and S Denote the Consumption and Social Link, Respectively

Model	Data Source			Prediction?		Evolution
	C	S	Time	C	S	Explanation?
PMF [34]	√	×	×	√	×	×
TMF [44]	√	×	√	√	×	×
SocialMF [15]	√	√	×	√	×	×
ContextMF [20]	√	√	×	√	×	×
AA [2]	×	√	×	×	√	×
CMF [9]	×	√	√	×	√	×
hTrust [40]	√	√	×	×	√	×
SAN [13]	√	√	×	×	√	×
FIP [45]	√	√	×	√	√	×
EJP [42]	√	√	√	√	√	√
EDCP	√	√	√	√	×	√
EDLP	√	√	√	×	√	√
ELCP	√	√	√	√	×	√
ELLP	√	√	√	×	√	√
ELJP	√	√	√	√	√	√

prediction. However, SAN does not perform well on this task. We guess a possible reason is that the consumption data is extremely sparse for both datasets, thus directly transforming the items users consume into the attributes would not produce good results. Second, among our proposed models, ELCP generates better result than EDCP, showing it is more effective to use latent representations to model user behavior. Also, ELJP always performs the best. This shows it is effective to jointly model users' two kinds of behaviors from an evolutional perspective. Furthermore, as EJP only modeled user latent consumption vector without any latent social representation, the results between EJP and ELJP demonstrates that it is more effective to jointly model user latent consumption vector and latent social vector. Last but not least, as the latent dimension increases, the performance improvement for all latent-based models is significant from $D=5$ to 10, and this improvement changes slowly after the latent dimension further increases. Given this observation, we set $D=10$ in the following experiments.

6.3 Social Link Prediction

We report link prediction results with:

- AA: The *A*damic/*A*dar metric is a c neighborhood-based unsupervised measure for link prediction [2].
- CMF: This temporal link prediction model collapsed multiple time series data into a single matrix, and then considered using matrix factorization based methods for predicting future links [9].
- hTrust: This proposed link prediction model exploited the homophily effect in link prediction via homophily regularization [40].
- SAN: This method augmented the social network into a social-attribute network to integrate network structure and node attributes to perform both link prediction and attribute inference [13].
- FIP: This method jointly modeled users' preference and social link prediction in a unified framework [45]

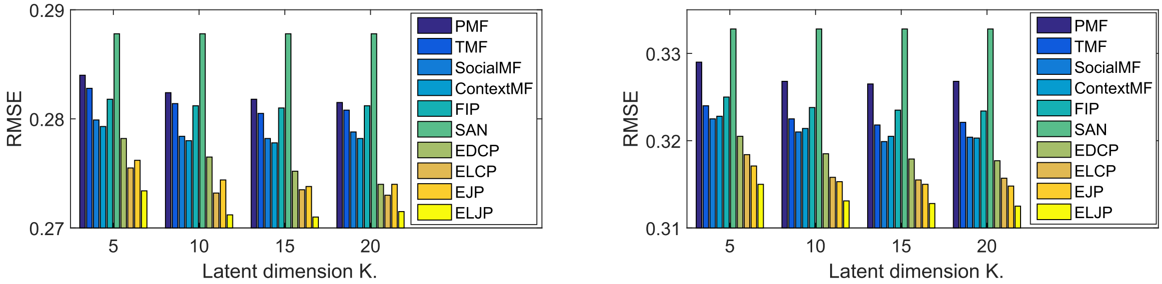
In link prediction task, our goal is to rank the potential linked users. As the user size is huge, it is impractical to take all users as candidate friends. Thus we adopt a similar approach that has been accepted by many works [22], [45]: for each test user a, we randomly sample 100 negatively linked users that are not connected to her till the test time window. Then we mix those positively linked users and the sampled negatively linked users together to select the top potential linked users of each test user. This process is repeated 10 times and we report the average results of all metrics. Particularly, we adopt three widely used *top-n* ranking metrics: *precision*, *recall*, and *F1* measure, where n denotes the size of the link prediction list [40], [45]. We set $n=5$ as it is useless to recommend too many friends, also, most online social networks adopt a similar number of potential friends for recommendation. In our direct link prediction model EDLP, we need to determine function h that predicts users' future link behavior based on the social network. Here, we choose the best outputs from AA and CMF since these two baselines only utilize the social link structure for link prediction.

Figs. 4 and 5 show the comparison metrics of these link prediction models on the two datasets. The latent dimensions are set to be $D=[5,10,15,20]$. Based on the results of the two datasets, AA performs better than CMF on Epinions data while CMF has better performance on Gowalla data. We guess a possible reason is that the Epinions data is much denser than the Gowalla data (i.e., 0.35 percent compared to 0.053 percent as shown in Table 1), thus the AA baseline can find more reliable potential social neighbors based on the denser social link structure. The hTrust baseline performs better than all purely link-based models and our proposed direct model EDLP has comparable results to hTrust, showing it is effective to leverage the homophily effect for link prediction. The comparison between our proposed models have similar trends as the results of consumption prediction. Particularly, the performance improvement of ELJP is significant over all other models. On average, the $F1$ measure improvement of ELJP over the best baseline is about 30 percent on Epinions and 20 percent on Gowalla. Note that besides $n=5$, we have also measured the link prediction performance with other values of n (from $n=1$ to $n=20$) and we found the overall trend is the same. Therefore, we do not report the detailed results at other settings of n. These results empirically validate it is reasonable and effective to jointly model users' consumption and link behaviors with latent representations from an evolving perspective.

Given the prediction results of the user consumption behavior and social link behavior, we conclude that the social network information and the user consumption behavior are mutually helpful, thus jointly modeling them from an evolving perspective would benefit both tasks. Among our proposed models, the ELJP always have the best performance. Compare ELJP with our proposed direct models, we conclude that it is more effective to model the evolution of users' behaviors with latent representations than direct representations. As the EJP model only represent the user with latent consumption vector while ELJP represents each user with a latent consumption vector and a latent social vector, the results empirically demonstrates that ELJP has better predictive power.

6.4 Computational Performance

Also, we compare the computational performance of all models. For fair comparison, we run all algorithms on the same platform. Since most algorithms need to iteratively calculate the parameters in each iteration, we list both the



(a) Epinions (b) Gowalla

Fig. 3. The overall comparison results of consumption prediction. The smaller the RMSE value, the better the prediction result.

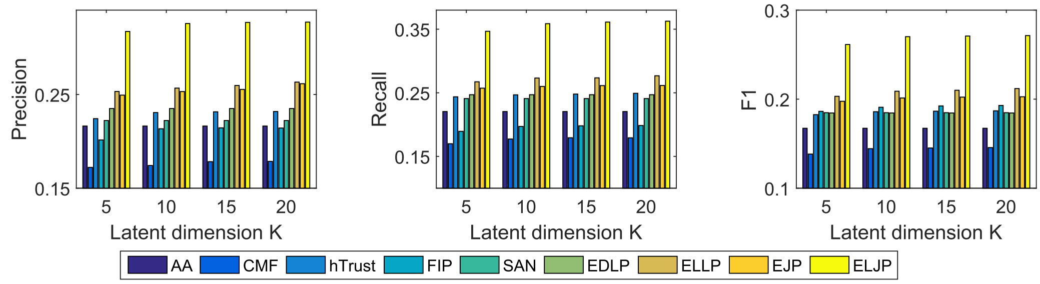


Fig. 4. The overall link comparison results on Epinions dataset. For all ranking metrics, the larger the value, the better the performance.

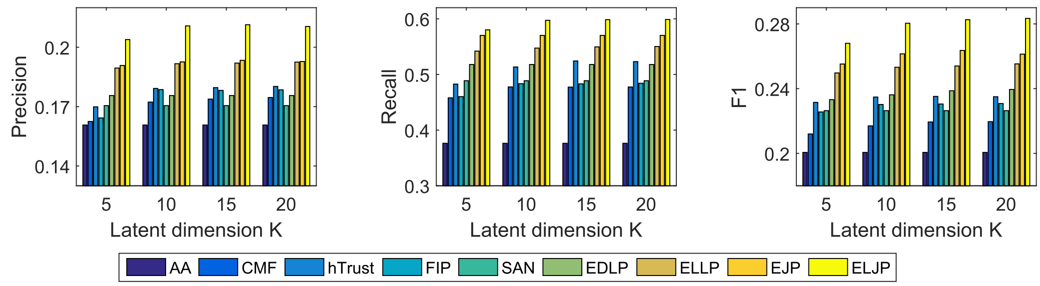
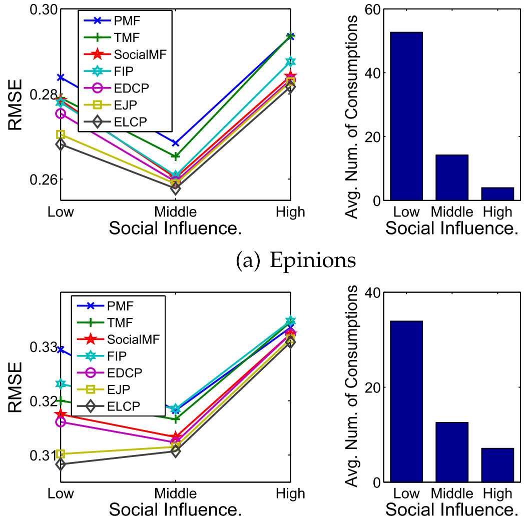


Fig. 5. The overall link comparison results on Gowalla dataset. For all ranking metrics, the larger the value, the better the performance.

TABLE 3
The Runtime of All Models (Min.) (For Each Dataset, the First Horizontal Line Shows the Runtime of Each Iteration and the Second Line Displays the Total Runtime)

Dataset	Time	Consumption Models						Link Models				Joint Models			
		PMF	TMF	SocailMF	ContextMF	SAN	EDCP	AA	CMF	hTrust	SAN	EDLP	FIP	EJP	ELJP
Epinions	Each	0.12	0.26	0.16	0.15	/	0.01	/	0.26	0.19	/	0.01	0.19	0.79	1.3
	Total	14	27	17	19	12	32	45	22	27	49	61	20	84	140
Gowalla	Each	0.68	1.1	0.8	0.9	/	0.08	/	0.84	0.98	/	0.08	1.5	1.35	5.02
	Total	72	114	84	97	43	119	1750	84	100	1930	146	150	142	510

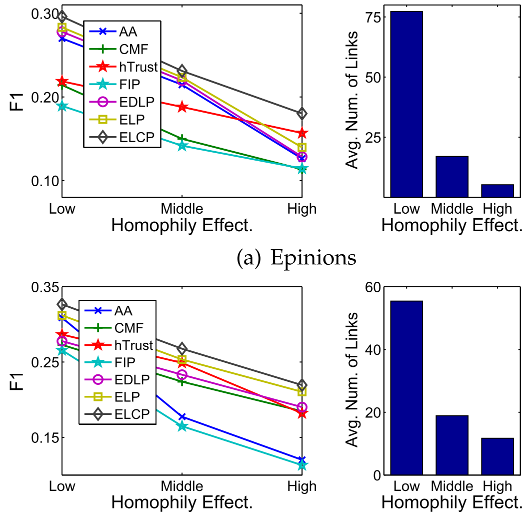
runtime of each iteration and the total runtime in Table 3. We set the iterations of all latent based models to be 100 as most algorithms' results converge after this number of iterations. As can be seen from this table, for consumption prediction models, in each iteration, the EDCP costs the least time as it only needs to calculate the balance parameters (Algorithm 1). PMF ranks the second as it is a base latent model for user consumption prediction. Compared to PMF, TMF and SocialMF need more runtime by adding the time and social network information. ELJP costs the most runtime in each iteration compared to the remaining models. With regard to the total runtime, the main time cost of our proposed EDCP model lies in calculating the direct social influence function g with limited consumption data. As the consumption data is usually very sparse, the total runtime of EDCP is still much less than ELJP. The similar runtime comparison can be also found in link prediction results, with CMF has the least time cost. On average, ELJP has as much as T times computational cost as the baselines such as SocialMF and hTrust. The reason is that ELJP needs to compute each user's latent consumption vector and latent social vector at each time slice, thus the time complexity is proportional to the time slices T as analyzed in Section 5.4. Given both the effectiveness and efficiency of the proposed models, we argue that our proposed direct model (EDCP and EDLP) has less time cost and comparable performance



(a) Epinions

(b) Gowalla

Fig. 6. The overall consumption performance over different social influence.



(a) Epinions

(b) Gowalla

Fig. 7. The overall link performance over different homophily effect.

while the proposed latent based models (EJP and ELJP) have much better performance than all baselines at a cost of runtime. But as discussed before, the complexity of ELJP is still linear with the number of consumption and social link records. Thus for real-world applications, we could train ELJP offline, and store the consumption and link predictions based on the output of ELJP in the server. Then in the online stage, users could get real-time predictions by retrieving the predictions from the server, which is time-efficient and can be applied to real-world SNSs.

6.5 Analyzing the Balance Parameters

Our proposed models can capture each user's unique preference for balancing the social influence and the homophily effect for her future decisions, i.e., α_a and β_a of each user a. We next study the relationship between the balance parameters and the prediction results. Particularly, as our proposed latent model ELJP has better performance than the proposed direct models, we analyze the balance parameters learned from ELJP. Specifically, we bin users into three equally sized groups according to the social influence effect value α_a, i.e., each user is grouped into the low, middle or high social-influence group. Then we compare the user consumption prediction results of each group in Fig. 6. On both datasets, nearly all models show the best performance on users that belong to middle social-influence group. And the low social-influence group usually has the worst performance among these three groups. To further investigate the reasons why this phenomenon happens, we depict the average number of consumption records on each group of users in the right part of the figure. On both datasets, the low social-influence group has the average largest number of consumption records, followed by the middle social influence group. The high social influence group has the least number of consumption records. On average, each user that belongs to the high social influence group only has 3.95 and 7.11 consumption records on Epinions and Gowalla respectively. These findings help us to explain a possible reason for the experimental discovery. If a user has very few consumption records, she is probably inexperienced and likely to be influenced by her social friends, leading them belong to the high social-influence group in our proposed latent model ELJP. Furthermore, the consumption prediction performance of this high social-influence group is low as ELJP does not have enough data to learn a user's consumption preference. On the contrary, as users have consumed more items, they are experienced in making consumption decisions with less influence from their social friends. The effectiveness on preference prediction also increases from high social-influence group to middle social-influence group as ELJP enjoys more consumption records to learn users' preferences. Nevertheless, the performance decreases from the middle social-influence group users to the low group. A possible reason is that users with more consumption records usually have consumed some infrequent items, thus are harder to predict. In fact, this decreasing trend of users that have too many consumption records has also been discovered by Wang et al. [41].

We also use similar techniques to group users into low, middle and high homophily groups according to the homophily effect value β_a, i.e., users in the low homophily groups have the smallest homophily values learned from ELJP. Fig. 7 shows the link prediction results of all models and the average link records under different groups. We choose $F1$ for the link prediction performance measure as it balances the measures of precision and recall. From Fig. 7, the correlation between the homophily values and the number of links has a similar trend as the social influence groups, with users of larger homophily effect values usually have less training records. The overall link prediction results for nearly all models increase as the homophily values decrease, as users in the low homophily group usually have more training records in model learning process.

6.6 Parameter Setting

Parameters in the Direct Model. In the proposed direct models of EDCP and EDLP, there are two parameters: the traditional collaborative filtering function f and social link prediction function h. In Tables 4 and 5, we show the performance of EDCP and EDLP with different base functions of f and h. As can be seen both tables, the direct prediction results perform better than the base functions. This suggests the advantage of the proposed direct models

TABLE 4
The RMSE Measure on EDCP over Different Collaborative Filtering Models

Dataset	Base model	f_{c2c}	EDCP	Improvement
Epinions	PMF	0.2824	0.2779	1.59%
	TMF	0.2810	0.2765	1.60%
Gowalla	PMF	0.3270	0.3240	0.93%
	TMF	0.3225	0.3185	1.24%

TABLE 5
The $F1$ Measure on EDLP over Different Link Prediction Models

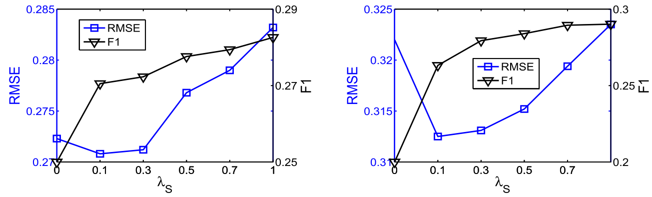
Dataset	Base model	h	EDCP	Improvement
Epinions	AA	0.1673	0.1845	10.28%
	CMF	0.1444	0.1592	10.24%
Gowalla	AA	0.2006	0.2169	7.51%
	CMF	0.217	0.2362	8.84%

compared with traditional consumption prediction and link prediction models. Also, the final prediction performance heavily relied on the choice of the base functions. The better performance of the base functions, the better final prediction results of the direct models. This finding is quit intuitive as both functions are integral parts of the final prediction results (as shown in Eqs. (1) and (4)). Nevertheless, due to the different characteristics of the two datasets, the best traditional baseline varies for different datasets. E.g., as shown in Table 5, the best traditional link baseline is CMF for Epinions and AA for Gowalla dataset. In summary, in order to get the best performance of our proposed direct models, we need to choose the best traditional model that suits the current data.

Parameters in the Latent Model. There are four parameters in our proposed latent model ELJP: λ_{U1}, λ_V, λ_S and λ_U. These parameters are important but not difficult to tune. Among them, λ_{U1} and λ_V are the regularization parameters of users' latent factors at time 1 and the item latent factor. Since these two parameters have a similar form as the traditional PMF model [34], we tune them on PMF and set them under the setting of the best performance on PMF. Thus we do not report the detailed setting of these two parameters. In the following, we report the setting of the remaining two parameters. Particularly, we choose the RMSE measure and F1 measure to evaluate the performance of these two tasks.

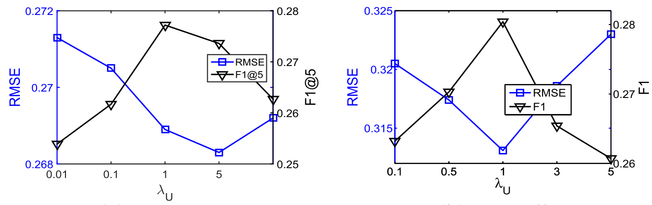
The setting of λ_S is shown in Fig. 8. For each λ_S, we initialize ELJP with random values, and stop model learning when either prediction task performance begins to decrease. In this figure, as λ_S increases from 0.1 to larger values, the overall trend is that the consumption performance decreases while the link performance increases as we put more weight on the social network information. Please note that both behavior prediction performance increases as we set λ_S from 0 to a 0.1. We explained it before as there are mutual relationship between users' behaviors, thus jointly modeling them would have better results. Given the results, setting λ_S in a reasonable range would balance these two prediction tasks, e.g., λ_S in $[0.3, 0.5]$ in Epinions and $[0.10.3]$ in Gowalla.

λ_U regularizes users' latent preference change over time, Fig. 9 gives the performance with varying parameters of λ_U. We observe that the values of λ_U impacts both behavior prediction results. As λ_U increases, the performance of both



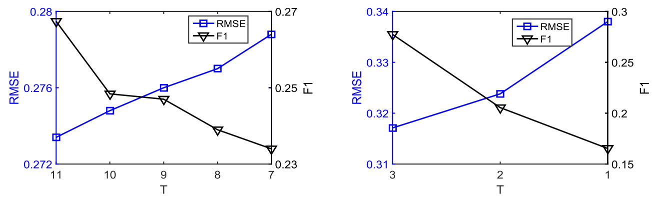
(a) Epinions (b) Gowalla

Fig. 8. The impact of λ_S.



(a) Epinions (b) Gowalla

Fig. 9. The impact of λ_U.



(a) Epinions (b) Gowalla

Fig. 10. The impact of temporal snapshots T.

prediction results increase at first, but when λ_U surpasses 5 in Epinions and 1 in Gowalla, the performance of the prediction results of both tasks decrease. Given this observation, we set $\lambda_U = 5$ in Epinions and $\lambda_U = 1$ in Gowalla data.

Parameter of temporal snapshots T. An important characteristic of our proposed problem is that it captures the temporal evolution of SNSs to model users' behaviors. To show the benefits modeling the temporal dynamics, we change the temporal snapshots size T and compare the results. Specifically, we use the last snapshot as the testing data and treat the previous T snapshots for training. We choose ELJP as a representative of our proposed models since it achieves the best performance. Fig. 10 gives the performance with varying parameters of T on the ELJP model. Note that since we only have 4 snapshots of Gowalla data, the maximum T equals 3 on this dataset. As can be seen from this figure, on both datasets, as T decreases, the consumption prediction performance and the social link prediction performance decrease. This observation results empirically shows the benefits of modeling temporal evolution of SNSs. As we add more temporal snapshots of SNSs, we could better capture the evolution of SNSs and make better prediction results.

7 Conclusion

We provided a focused study on understanding and modeling users' temporal behaviors in SNS platforms. Particularly, we proposed two representations to depict the evolution of users' temporal behaviors in SNS platforms: a direct representation that presumes users' behaviors are represented directly by their historical behaviors and a latent representation that assumes users' behaviors are encoded latently from the observable behaviors. For each representation, we associated users' two kinds of behaviors with two time-variant vectors. Furthermore, for each representation, we provided the corresponding models to incorporate the underlying social theories for users' evolving behaviors, where the social influence and homophily effect for users' behaviors are clearly quantified. Thus our proposed models have both the predictive power and the explanation ability from an individual perspective. Experimental results validated that the users' preferences and the social network information are mutually helpful, thus jointly modeling them would benefit both user consumption prediction and the social link prediction task. In the future, on one hand, we would like to follow this direction and explore how to build a more effective SNS platform based on our findings. On the other hand, as users' behavior data comes from time to time, how to incrementally update our proposed models to apply it in the real world SNSs is another interesting research direction.

Acknowledgments

This work was supported in part by grants from the National 973 Program of China (Grant No. 2013CB329604 and 2013CB336500), the Nature Science Foundation of China (Grant No. 61602147, 61432019, 61403358, 61472116 and U1605251), the National Science Foundation for Distinguished Young Scholars of China (Grant No. 61325010), and the Key Program of the Nature Science Foundation of China (Grant No. 61532006). This paper was an expanded version of [42], which appeared in *the Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI 2016)*. Enhong Chen is the corresponding author.

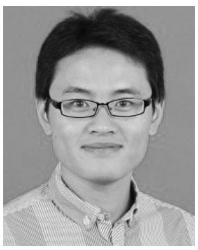
References

[1] E. Acar, D. M. Dunlavy, and T. G. Kolda, "Link prediction on evolving data using matrix and tensor factorizations," in *Proc. IEEE Int. Conf. Data Mining Workshops*, 2009, pp. 262–269.
[2] L. A. Adamic and E. Adar, "Friends and neighbors on the web," *Social Netw.*, vol. 25, no. 3, pp. 211–230, 2003.
[3] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 6, pp. 734–749, Jun. 2005.
[4] S. Aral, L. Muchnik, and A. Sundararajan, "Distinguishing influence-based contagion from homophily-driven diffusion in dynamic networks, " *Proc. Nat. Academy Sci. United States America*, vol. 106, no. 51, pp. 21544–21549, 2009.
[5] L. Bottou, "Large-scale machine learning with stochastic gradient descent, " in *Proc. COMPSTAT*, Berlin, Germany: Springer, 2010.
[6] J. S. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," in *Proc. 14th Conf. Uncertainty Artif. Intell.*, 1998, pp. 43–52.
[7] D. Crandall, D. Cosley, D. Huttenlocher, J. Kleinberg, and S. Suri, "Feedback effects between similarity and social influence in online communities," in *Proc. 14th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2008, pp. 160–168.
[8] Y. Ding and X. Li, "Time weight collaborative filtering," in *Proc. 14th ACM Int. Conf. Inf. Knowl. Manage.*, 2005, pp. 485–492.
[9] D. M. Dunlavy, T. G. Kolda, and E. Acar, "Temporal link prediction using matrix and tensor factorizations," *ACM Trans. Knowl. Discovery Data*, vol. 5, no. 2, 2011, Art. no. 10.
[10] N. E. Friedkin, *A Structural Theory of Social Influence*. vol. 13, Cambridge, U.K.: Cambridge Univ. Press, 2006.
[11] S. Gao, L. Denoyer, and P. Gallinari, "Temporal link prediction by integrating content and structure information," in *Proc. 20th ACM Int. Conf. Inf. Knowl. Manage.*, 2011, pp. 1169–1174.
[12] D. Goldberg, D. Nichols, B. M. Oki, and D. Terry, "Using collaborative filtering to weave an information tapestry," *Commun. ACM*, vol. 35, no. 12, pp. 61–70, 1992.
[13] N. Z. Gong, et al., "Joint link prediction and attribute inference using a social-attribute network," *ACM Trans. Intell. Syst. Technol.*, vol. 5, no. 2, 2014, Art. no. 27.
[14] M. Jamali and M. Ester, "TrustWalker: A random walk model for combining trust-based and item-based recommendation," in *Proc. 15th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2009, pp. 397–406.
[15] M. Jamali and M. Ester, "A matrix factorization technique with trust propagation for recommendation in social networks," in *Proc. 4th ACM Conf. Recommender Syst.*, 2010, pp. 135–142.
[16] M. Jamali, G. Haffari, and M. Ester, "Modeling the temporal dynamics of social rating networks using bidirectional effects of social relations and rating patterns," in *Proc. 20th Int. Conf. World Wide Web*, 2011, pp. 527–536.
[17] G. Jeh and J. Widom, "SimRank: A measure of structural-context similarity," in *Proc. 8th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2002, pp. 538–543.
[18] M. Jiang, et al., "Social contextual recommendation," in *Proc. 21st ACM Int. Conf. Inf. Knowl. Manage.*, 2012, pp. 45–54.
[19] M. Jiang, P. Cui, F. Wang, X. Xu, W. Zhu, and S. Yang, "FEMA: Flexible evolutionary multi-faceted analysis for dynamic behavioral pattern discovery," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2014, pp. 1186–1195.
[20] M. Jiang, P. Cui, F. Wang, W. Zhu, and S. Yang, "Scalable recommendation with social contextual information," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 11, pp. 2789–2802, Nov. 2014.
[21] D. Kempe, J. Kleinberg, and É. Tardos, "Maximizing the spread of influence through a social network," in *Proc. 9th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2003, pp. 137–146.
[22] Y. Koren, "Factorization meets the neighborhood: A multifaceted collaborative filtering model," in *Proc. 14th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2008, pp. 426–434.
[23] Y. Koren, "Collaborative filtering with temporal dynamics," *Commun. ACM*, vol. 53, no. 4, pp. 89–97, 2010.
[24] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Comput.*, vol. 42, no. 8, pp. 30–37, Aug. 2009.
[25] P. F. Lazarsfeld and R. K. Merton, "Friendship as a social process: A substantive and methodological analysis," *Freedom Control Modern Soc.*, vol. 18, no. 1, pp. 18–66, 1954.
[26] J. Leskovec, D. Huttenlocher, and J. Kleinberg, "Predicting positive and negative links in online social networks," in *Proc. 19th Int. Conf. World Wide Web*, 2010, pp. 641–650.
[27] H. Li, Y. Ge, R. Hong, and H. Zhu, "Point-of-interest recommendations: Learning potential check-ins from friends," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2016, pp. 975–984.
[28] D. Liben-Nowell and J. Kleinberg, "The link-prediction problem for social networks," *J. Amer. Soc. Inf. Sci. Technol.*, vol. 58, no. 7, pp. 1019–1031, 2007.
[29] C.-b. Lin, "Projected gradient methods for nonnegative matrix factorization," *Neural Comput.*, vol. 19, no. 10, pp. 2756–2779, 2007.
[30] Q. Liu, E. Chen, X. Hui, C. H. Ding, and J. Chen, "Enhancing collaborative filtering by user interests expansion via personalized ranking," *IEEE Trans. Systems Man Cybernetics, Part B (Cybernetics)*, vol. 42, no. 1, pp. 218–233, Feb. 2012.
[31] M. McPherson, L. Smith-Lovin, and J. M. Cook, "Birds of a feather: Homophily in social networks," *Annu. Rev. Sociology*, vol. 27, pp. 415–444, 2001.
[32] A. K. Menon and C. Elkan, "Link prediction via matrix factorization," in *Proc. 2011 Eur. Conf. Mach. Learn. Knowl. Discovery Databases*, 2011, pp. 437–452.
[33] K. Miller, M. I. Jordan, and T. L. Griffiths, "Nonparametric latent feature models for link prediction," in *Proc. 22nd Int. Conf. Neural Inf. Process. Syst.*, 2009, pp. 1276–1284.
[34] A. Mnih and R. Salakhutdinov, "Probabilistic matrix factorization," in *Proc. Conf. Neural Inf. Process. Syst.*, 2007, pp. 1257–1264.
[35] S. Rendle, C. Freudenthaler, and L. Schmidt-Thieme, "Factorizing personalized markov chains for next-basket recommendation," in *Proc. 19th Int. Conf. World Wide Web*, 2010, pp. 811–820.
[36] M. Richardson, R. Agrawal, and P. Domingos, "Trust management for the Semantic Web," in *Proc. 2nd Int. Conf. Semantic Web Conf.*, 2003, pp. 351–368.
[37] P. Sarkar, D. Chakrabarti, and M. Jordan, "Nonparametric link prediction in dynamic networks," in *Proc. 29th Int. Conf. Mach. Learning*, 2012, pp. 1687–1694.

[38] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," in *Proc. 10th Int. Conf. World Wide Web*, 2001, pp. 285–295.
[39] S. Scellato, A. Noulas, and C. Mascolo, "Exploiting place features in link prediction on location-based social networks," in *Proc. 17th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2011, pp. 1046–1054.
[40] J. Tang, H. Gao, X. Hu, and H. Liu, "Exploiting homophily effect for trust prediction," in *Proc. 6th ACM Int. Conf. Web Search Data Mining*, 2013, pp. 53–62.
[41] C. Wang and D. M. Blei, "Collaborative topic modeling for recommending scientific articles," in *Proc. 17th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2011, pp. 448–456.
[42] L. Wu, Y. Ge, Q. Liu, E. Chen, B. Long, and Z. Huang, "Modeling users' preferences and social links in social networking services: A joint-evolving perspective," in *Proc. 13th AAAI Conf. Artif. Intell.*, 2016, pp. 279–286.
[43] B. Xiang, Q. Liu, E. Chen, H. Xiong, Y. Zheng, and Y. Yang, "PageRank with priors: An influence propagation perspective," in *Proc. 23rd Int. Joint Conf. Artif. Intell.*, 2013, pp. 2740–2746.
[44] L. Xiong, X. Chen, T.-K. Huang, J. G. Schneider, and J. G. Carbonell, "Temporal collaborative filtering with Bayesian probabilistic tensor factorization," in *Proc. SIAM Int. Conf. Data Mining*, 2010, vol. 10, pp. 211–222.
[45] S.-H. Yang, B. Long, A. Smola, N. Sadagopan, Z. Zheng, and H. Zha, "Like like alike: Joint friendship and interest propagation in social networks," in *Proc. 20th Int. Conf. World Wide Web*, 2011, pp. 537–546.

Le Wu received the PhD degree in computer science from the University of Science and Technology of China. She is currently a faculty member with the Hefei University of Technology, China. Her general area of research is data mining, recommender system, and social network analysis. She has published several papers in referred conference proceedings and journals, such as AAAI, SIAM SDM, IEEE ICDM, ACM CIKM, and the *ACM Transactions on Intelligent Systems and Technology*. She received the Best of SDM 2015 Award.

Yong Ge received the PhD degree in information technology from the State University of New Jersey, Rutgers, in 2013. He is an assistant professor of management information systems with the University of Arizona. His research interests include data mining and business analytics. He received the ICDM-2011 Best Research Paper Award. He has published prolifically in refereed journals and conference proceedings, such as the *IEEE Transactions on Knowledge and Data Engineering*, the *ACM Transactions on Information Systems*, the *ACM Transactions on Knowledge Discovery* from Data, ACM SIGKDD, SIAM SDM, IEEE ICDM, and ACM RecSys. He also was the program committee member at ACM SIGKDD, IEEE ICDM, etc.



Qi Liu received the PhD degree in computer science from University of Science and Technology of China (USTC). He is an associate professor with the USTC. His general area of research is data mining and knowledge discovery. He has published prolifically in refereed journals and conference proceedings, e.g., the *IEEE Transactions on Knowledge and Data Engineering*, the *ACM Transactions on Information Systems*, the *ACM Transactions on Knowledge Discovery from Data*, the *ACM Transactions on Intelligent Systems and Technology*, KDD, IJCAI, AAAI, ICDM, SDM, and CIKM. He has served regularly on the program committees of a number of conferences, and is a reviewer for the leading academic journals in his fields. He received the ICDM 2011 Best Research Paper Award and the Best of SDM 2015 Award. He is a member of ACM and the IEEE.

Enhong Chen (SM'07) received the PhD degree from the University of Science and Technology of China. He is a professor and vice dean of the School of Computer Science, USTC. His general area of research includes data mining and machine learning, social network analysis, and recommender systems. He has published more than 100 papers in refereed conferences and journals, including *IEEE Transactions on Knowledge and Data Engineering*, *IEEE Transactions on Industrial Electronics*, KDD, ICDM, NIPS, and CIKM. He was on program committees of numerous conferences including KDD, ICDM, and SDM. He received the Best Application Paper Award on KDD-2008, the Best Research Paper Award on ICDM-2011, and the Best of SDM-2015. His research is supported by the US National Science Foundation for Distinguished Young Scholars of China. He is a senior member of the IEEE.

Richang Hong (M'12) received the PhD degree from the University of Science and Technology of China, Hefei, China, in 2008. He is currently a professor with the Hefei University of Technology. He was a research fellow in the School of Computing, National University of Singapore, as a research fellow from 2008 to 2010. He has co-authored more than 60 publications in the areas of his research interests, which include multimedia question answering, video content analysis, and pattern recognition. He is a member of the Association for Computing Machinery. He received the best paper award in the ACM Multimedia 2010. He is a member of the IEEE.

Junping Du received the PhD degree in computer science from the University of Science and Technology Beijing (USTB), and then held a postdoc fellowship in the Department of Computer Science, Tsinghua University, Beijing, China. She joined the School of Computer Science, Beijing University of Posts and Telecommunications (BUPT), in July 2006, where she is currently a professor of computer science. She served as chair and co-chair of IPC for many international and domestic academic conferences, and has been vice general secretary of the Chinese Association for Artificial Intelligence (CAAI) since 2004. She was a visiting professor in the Department of Computer Science, Aarhus University, Denmark, from September 1996 until September 1997. Her current research interests include artificial intelligence, data mining, intelligent management system development, and computer applications.

Meng Wang (M'09) received the BE and PhD degrees from the Special Class for the Gifted Young, Department of Electronic Engineering and Information Science, University of Science and Technology of China (USTC), Hefei, China, in 2003 and 2008, respectively. He is a professor with the Hefei University of Technology, China. His current research interests include multimedia content analysis, computer vision, and pattern recognition. He has authored more than 200 book chapters, and journal and conference papers in these areas. He was the recipient of the ACM SIGMM Rising Star Award in 2014. He is an associate editor of the *IEEE Transactions on Knowledge and Data Engineering*, the *IEEE Transactions on Circuits and Systems for Video Technology*, and the *IEEE Transactions on Neural Networks and Learning Systems*. He is a member of the IEEE.