Exploiting Implicit Influence from Information Propagation for Social Recommendation

Fei Xiong, Weihan Shen, Hongshu Chen, Shirui Pan, *Member, IEEE*, Ximeng Wang and Zheng Yan

***Abstract*—Social recommender systems have attracted a lot of attention from academia and industry. On social media, users' ratings and reviews can be observed by all users, and have implicit influence on their future ratings. When these users make subsequent decisions about an item, they may be affected by existing ratings on the item. Thus, implicit influence propagates among the users who rated the same items, and it has significant impact on users' ratings. However, implicit influence propagation and its effect on recommendation rarely have been studied. In this paper, we propose an information propagation-based social recommendation method (SoInp) and model the implicit user influence from the perspective of information propagation. The implicit influence is inferred from ratings on the same items. We investigate the concrete effect of implicit user influence in the propagation process and introduce it into recommender systems. Furthermore, we incorporate the implicit user influence and explicit trust information in the matrix factorization framework. To demonstrate the performance, we conduct comprehensive experiments on real-world datasets to compare the proposed method with state-of-the-art models. The results indicate that SoInp makes notable improvements in rating prediction.**

***Index Terms*—Computational Intelligence, Recommender Systems, Information Propagation, Implicit User Influence, Social Networks.**

I. INTRODUCTION

WITH the development of online social media, a huge amount of information and extensive applications on the Internet have influenced or even redefined our lives in many ways. However, enormous contents on the web make it difficult for users to obtain the information they need. In many situations, individual demands can be described by only a few search terms. Therefore, recommender systems have become an important way of filtering information to overcome the problem of information overload. Recommender systems have been used successfully in a wide variety of fields, such as websites for music [1], movies [2, 3] and social news [4]. Meanwhile, as social media have sprung up, online users are allowed to participate in many kinds of activities that produce large numbers of social relationships, such as friendships in Facebook and trust relationships in Epinions.

F. Xiong and W. Shen are with the Key Laboratory of Communication & Information Systems, Beijing Municipal Commission of Education, Beijing Jiaotong University, Beijing 100044, China. (E-mail: xiongf@bjtu.edu.cn, 17120108@bjtu.edu.cn)

H. Chen is with the School of Management and Economics, Beijing Institute of Technology, Beijing 100081, China. (E-mail: Hongshu.Chen@bit.edu.cn)

S. Pan is with the Faculty of Information Technology, Monash University, Clayton, VIC 3800, Australia. (E-mail: shirui.pan@monash.edu)

X. Wang is with the Key Laboratory of Communication & Information Systems, Beijing Municipal Commission of Education, Beijing Jiaotong University, Beijing 100044, China and also with the Centre for Artificial Intelligence, Faculty of Engineering and Information Technology, University of Technology Sydney, NSW 2007, Australia. (E-mail: wangxm@bjtu.edu.cn)

Z. Yan is with the Centre for Artificial Intelligence, Faculty of Engineering and Information Technology, University of Technology Sydney, NSW 2007, Australia. (E-mail: Yan.Zheng@uts.edu.au)

Manuscript received December 01, 2018, revised March 29, 2019.

Collaborative filtering (CF) is one of the most popular and effective techniques in recommender systems, and it can be divided into memory-based [5, 6] and model-based [7–11] categories. CF uses past reviews or rating data to make fine recommendation without the need for exogenous information. In CF, users with similar preferences in the past are supposed to agree on the same items in the future. However, CF suffers from two well-known issues, i.e., data sparsity and cold start [12, 13]. It is a fact that users usually rate or experience only a small fraction of available items. Cold-start users with few ratings pose a great challenge for CF methods. Therefore, traditional recommender systems that purely mine the data of ratings may not provide sufficiently reliable prediction.

To address these problems, many researchers have introduced social information into recommendation models [14–18]. In reality, users' decisions often are affected by their trusted friends. Since traditional recommender systems usually lack the capability of differentiating users' creditability, i.e., the trustworthiness of users' ratings, social information provides an independent source for recommendation. Social information and ratings can complement each other, and combining them is very beneficial in mining users' preferences. Therefore, more appropriate recommendations are provided to users.

For the reason stated above, trust-aware recommender systems have attracted a lot of attention [19, 20]. Trust- aware models assume that people usually prefer to consider the opinions of their trusted friends when making decisions or choices. Although trust relationships have a key role in improving recommender systems, the performance of many trust-aware methods may be inferior to those of some models which merely consider users' ratings [18]. One possible explanation is that these trust-aware models focus too much on the utility of users' trust and do not make full use of the influence of ratings. This result inspires us to make extensive use of potential information from ratings in addition to explicit values. User influence does not only originate from social networks, but also can be inferred by observed ratings on jointly-rated items.

Users on social media often are influenced by others to form opinions and take actions, resulting in the process of information diffusion. For instance, after a user accepts a message, other users may be affected and also will consider accepting the message. The message may be a tweet on

Twitter, a movie on a video website, or an item on a product review website. Information diffusion is caused by users' actions which include retweeting a tweet or rating an item. Therefore, the aim of personalized recommendation is to explore the acceptance of items over a population, and it can be regarded as an application of information diffusion [21]. For recommendation, after users' ratings and reviews have been published, they can be observed by all other users. When a user rates an item, he/she is affected by the users who rated the item, irrespective of whether or not they have direct links. The implicit influence propagates among the users who rated the same items. However, implicit influence propagation rarely has been studied, and its effect on recommendation still needs additional study. In this paper, we investigate the effect of implicit user influence based on information propagation, and we consider the influence in recommendation. Also, we propose a social recommendation method that incorporates the implicit user influence from ratings and explicit trust influence from social networks. Experimental results on two real-world datasets demonstrate that our method performs significantly better than state-of-the-art models. In addition, our method does not increase the computation complexity so it is suitable for practical use. The main contributions of this paper are listed below:

1) We investigate the implicit user influence from the perspective of information propagation. We analyze the concrete effect of user influence by the mean-field approach for two propagation mechanisms, i.e., linear threshold and independent cascade. The analysis on homogeneous and heterogeneous networks indicates the concrete effect of user influence.

2) We introduce the implicit user influence to the matrix factorization for recommender systems. Ratings on an item are correlated with users who rated the item in the past. We also revamp our recommendation method by incorporating both the implicit user influence and explicit trust influence to improve its performance.

3) We conduct comprehensive experiments to evaluate the accuracy of the proposed method (SoInp). We compare SoInp with state-of-the-art models on two real-world datasets. Results indicate that the implicit user influence based on information propagation has a significant role in recommendation, and SoInp makes a remarkable improvement in the task of rating prediction both for all users and cold-start users.

The rest of this paper is organized as follows. In Section II, we briefly review related work about recommender systems. Section III describes the proposed method with the implicit influence based on information propagation. Experimental settings and results on real-world datasets are reported in Section IV, and this is followed by some concluding remarks and an outline of our future work.

II. Related work

A. Matrix factorization

As a typical representative of model-based CF, matrix factorization (MF) is widely used in recommender systems [22]. It is an efficient and effective approach which maps the $m \times n$ rating matrix into two low-rank matrices [23, 24], where m denotes the number of users, and n denotes the number of items. This approach factorizes the user-item rating matrix and utilizes the generated user-specific and item-specific matrices to predict missing ratings. MF seeks to approximate the rating matrix $\boldsymbol{R} \in \mathbb{R}^{m \times n}$ by the inner product of two low-rank matrices, i.e., $\boldsymbol{U} \in \mathbb{R}^{d \times m}$ and $\boldsymbol{V} \in \mathbb{R}^{d \times n}$, as follows:

$$\boldsymbol{R} \approx \boldsymbol{U}^T \boldsymbol{V} \tag{1}$$

Traditionally, the aim of MF is to decompose the rating matrix by minimizing the sum-of-squared-errors objective function with quadratic regularization terms, as:

$$\mathcal{L} = \frac{1}{2} \sum_u \sum_{j \in I_u} (V_j^T U_u - r_{u,j})^2 + \frac{\lambda}{2} (\sum_u \|U_u\|^2 + \sum_j \|V_j\|^2) \tag{2}$$

where U_u is a d-dimensional vector ($d < \min(m, n)$) that represents user u's latent factor vector, V_j is the latent vector of item j, and I_u is the set of items which user u rated. $\lambda (\lambda > 0)$ is a trade-off parameter which controls the model complexity and avoids over fitting, and $\|*\|$ is the Euclidean norm. Gradient-based approaches can be used to solve the problem of minimizing the objective function. Then, missing ratings are predicted by the inner product, $\boldsymbol{U}^T \boldsymbol{V}$, of the user-specific and item-specific matrices.

B. Social recommendation

Original MF only utilizes ratings. In fact, it has been proved that social information and ratings can complement each other [18, 25, 26]. This finding inspires researchers to study trust-aware recommendation. In recent years, with the occurrence of abundant social data, trust-aware recommender systems have been presented extensively. Tan et al. [27] developed a novel music recommendation model that uses a variety of social information and music acoustic-based contents. In their model, users have more complex relationships than pairwise connections. Ma et al. [14] proposed a probabilistic MF method, i.e., SoRec, which factorizes both the rating matrix and social relationship matrix. Most of these existing methods only combine users' preferences with their trusted friends and seek to determine the impact of users' interactions in local regions. Besides explicit influence for real values of ratings and trust, implicit influence also helps improve the recommendation. The implicit trust means that users' decisions are affected by friends' ratings, and the implicit trust can be inferred from ratings. Guo et al. [28] proposed a trust-aware MF model named TrustSVD which incorporates both the explicit and implicit influence of trust for the prediction of missing ratings. TrustSVD outperforms most of state-of-the-art social recommendation models.

Users' relationships on social networks are complex and diverse. Behaviors of users usually are affected by their social relationships which involve directly and indirectly connected friends. Yang et al. [29] studied interest propagation on social networks and found that information extracted from interest networks and friendship networks was truly relevant and mutually helpful. Therefore, they proposed a friendship-interest propagation model to predict friendships, and the propagation was applied for social recommendation. Massa and Avesani

[30] proposed a trust-aware recommender system that used a trust metric to propagate trust over a trust network instead of finding similar users, since the similarity calculation often failed due to data sparsity of input ratings.

In addition to the influence on friends' preferences, more information from trust relationships has been used to improve the recommendation. Golbeck [2] presented an approach which allows users to encode more information about their relationships rather than just stating the people they know or trust. In [31], trust propagation enhances the coverage of recommender systems while simultaneously maintaining recommendation accuracy and quality. In this approach, the CF process is determined by users' reputation. Guha et al. [32] proved that minority trust/distrust relationships contributed to predict the trust between any pairwise users. Parvin et al. [33] incorporated trust information in a non-negative MF framework to solve cold-start problems. In [34], each user's decisions depend on the ratings of the users who have direct or indirect social relationships with him/her. In [35], trust statements along with raings were treated as side information in a nonnegative MF framework.

Item recommendation can be regarded as the prediction of acceptance for items over a population in the process of information propagation so that studies on information propagation can contribute to recommender systems. Duo et al. [36] proposed a unique form of user similarity, i.e., transferring similarity, which considers all high-order similarities among users. In their model, the greater the distance between two users is, the smaller their impact is.

Table I shows the main properties of the methods we mentioned above. In the table, d means the number of latent factors, $|\boldsymbol{R}|$ means the number of ratings, $|\boldsymbol{C}|$ means the number of relationships, m is the number of users, n is the number of items, S means the average number of direct relationships per user, and $l = max(M, k^{+}, k^{-})$ where M is the average number of ratings received by an item, k^{+} and k^{-} are the average number of trust statements given and received, respectively, by a user. In summary, trust-aware models open a research direction of recommender systems. In addition to trust networks, the ratings themselves imply implicit user influence which affects users' decisions in the future. The implicit influence propagates among the users who rated the same items. However, implicit influence propagation rarely has been studied. In this paper, we model the implicit influence of users from the perspective of information propagation, and we connect explicit ratings, social trust relationships and implicit user influence together.

III. Recommendation with Implicit Influence Based on Information Propagation

In this section, we explore how users are influenced by others when they make a decision, and we introduce the concept of information propagation to recommender systems. Figure 1 shows the process of information propagation. Typically, a user's rating on item j is influenced to different extents by the users who rated item j. If the user gives a rating on item j, it has a certain effect on the users who will rate item j in the future.

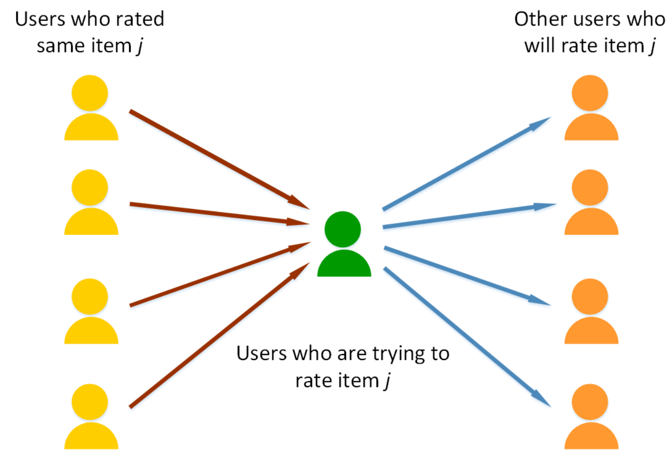


Fig. 1. The process of information propagation.

In the following sections, we provide more details concerning the motivations of our work in consideration of information propagation in recommender systems. Then, we describe two mechanisms of information propagation among users. Our detailed method is introduced in Section III-C. The explicit influence of trust relationships is addressed in Section III-D and the training algorithm of the proposed method is given in Section III-E. In Section III-F, we analyze the computational complexity of the proposed method. Table II shows the main notations defined in this paper.

A. Motivations

There are kinds of social relationships which affect users' behaviors or decisions, such as friends, colleagues and schoolmates. However, users do not always accept recommendations that are provided by each friend, implying that the influence of these friends is different. In addition, each user also is affected by the users who rated the same items with him/her, even if there is no direct connection between them. The objective of this paper is to effectively predict missing values in the user-item rating matrix by integrating information from multiple data sources. We model the implicit influence of users who rated the same items from the perspective of information propagation, and we incorporate explicit ratings with implicit user influence.

The motivations of this paper are described as follows:

1) Existing social recommender systems often assume that users' preferences are influenced by direct friends. In fact, implicit user influence also comes from the people who rated the same items. For instance, when user u publishes a rating and some comments on an item, besides the rating, the comments and user information also can be noticed by other users. When these users make decisions on the same item, they may be affected by user u. These decisions also influence other users' future ratings on the item. Therefore, user influence propagates among these users. However, implicit influence propagation rarely has been considered in recommendation.

2) User influence depends on specific items. A user may be an expert on an item, but he/she may have no knowledge about another item. Therefore, influence between two users is

TABLE I
ANALYSIS OF THE MAIN PROPERITES OF METHODS

Methods	Recommendation type	Side information	Cold-start control	Time complexity in an iteration
MRH[27]	Model-Based	Trust	✓	-
SoRec[14]	Model-Based	Trust	✓	$O(\|\boldsymbol{R}\|d + \|\boldsymbol{C}\|d)$
TrustSVD[28]	Model-Based	Trust	✓	$O(d\|\boldsymbol{R}\|l + d\|\boldsymbol{C}\|l)$
FIP[29]	Model-Based	Trust	×	-
TrustAR[30]	CF	Trust	✓	-
TBR[2]	CF	-	×	-
TrustRSNF[35]	Model-Based	Trust	×	$O(d * max(m, n))$
TrustMF[34]	Model-Based	Trust	×	$O(mMd + mSd)$
TrustANLF[33]	Model-Based	Trust	✓	$O(d^2\|\boldsymbol{R}\|)$

TABLE II
NOTATIONS

Symbol	Description
C_u	the set of user u's neighbor
δ_v	the state of user v in information propagation
$P_{u,v}$	the probability of inactive user u being activated by active user v
ε	the average probability of being activated by an active neighbor
$\rho_a(t)$	the proportion of active users at time t in homogeneous networks
$\rho_i(t)$	the proportion of inactive users at time t in homogeneous networks
$P(k^{'}\|k)$	the probability that a user with degree k connects to a user with degree $k^{'}$
$\rho_a(k,t)$	the proportion of active users with degree k at time t
U_u	the latent vector of user u
W_v	the latent vector of trustee v
V_j	the latent vector of item j
$\hat{r}_{u,j}$	the prediction of u's rating score on item j
$\hat{c}_{u,v}$	the predicted trust relationship between user u and v
I_u	the set of items which user u rated
b_u	inherent bias of user u
b_j	inherent bias of item j
μ	global average rating
Γ_j	the set of users who rated item j
y_i	the implicit influence vector of user i who rated item j
α	a constant which controls the extent of implicit influence
λ_t	a constant which controls the extent of trust regularization
λ	the regularization parameter for latent factors
d	the number of latent factors

different towards different items. User influence for a target item is correlated with latent factors of the item.

3) Information propagation shows how a message is disseminated from a user to others. The probability of accepting a message from another user depends on the user's influence. A message can be treated as an item in recommender systems, and accepting and using the message is regarded as publishing a rating. Therefore, the implicit user influence in recommender systems can be derived from the process of information propagation, and it can be introduced to the MF framework.

B. Information propagation

On social media, once a user publishes a message, the message is transmitted to his/her neighbors. The decision of whether to accept and repost the message is correlated with the content of the message and the influence of the author. When neighbors repost the messages, more users have an opportunity to read it. Then, the message propagates beyond the original author's local region. There are dynamic models that simulate information propagation among users. At the beginning of the dynamics for these models, one or several users are set in the active state, and the others remain inactive. The active state means users have accepted the message and tried to diffuse it, while inactive users do not have any contact with it. Inactive users may be activated by active neighbors with different probability. Here, we focus on two basic mechanisms of information propagation, i.e., linear threshold [37] and independent cascade [38].

Linear threshold: Assume that there is a social network represented by a graph which includes the set of users and the set of social relationships. Each user chooses a threshold θ_u which is selected at random from a uniform distribution in the interval $[0,1]$. In each iteration, inactive user u becomes active if the weighted sum of the edges with active neighbors exceeds the threshold θ_u, shown as follows:

$$\sum_{v\in C_u}\delta_v P_{u,v}\geq\theta_u \tag{3}$$

where C_u is the set of user u's neighbors, and $\delta_v=1$ if user v is active, otherwise $\delta_v=0$. Inactive user u is activated by active user v with probability $P_{u,v}$. The aforementioned activation condition is not differentiable, so that the process of information propagation is hard to analyze by the mean-field approach. Therefore, we generalize the activation probability of inactive user u by linear superposition of active neighbors, as:

$$P_{u,i\to a}=\sum_{v\in C_u}\delta_v P_{u,v}/|C_u| \tag{4}$$

Independent cascade: In this case, if user u has some neighbors, he/she is activated by each neighbor independently with different probability. One can infer the activation probability of user u as:

$$P_{u,i\to a}=1-\prod_{v\in C_u}(1-\delta_v P_{u,v}) \tag{5}$$

The two mechanisms of information propagation shown above can be extended to continuous-time dynamics. Given a continuous time interval τ, for the propagation of linear superposition or independent cascade, the activation probability of inactive users during τ can be obtained as follows.

We introduce a small auxiliary variable Δt. Users can diffuse the message in Δt, so that inactive user u is activated by active neighbors in Δt with probability $P_{u,i\to a}\Delta t$. Therefore, the activation probability of user u during τ is:

$$P_u^\tau=1-\lim_{\Delta t\to 0}(1-P_{u,i\to a}\Delta t)^{\tau/\Delta t}=1-\mathrm{e}^{-P_{u,i\to a}\tau} \tag{6}$$

Therefore, the activation probability during continuous time accumulates as an exponential function.

C. Recommendation based on information propagation

Users' ratings have a correlation with the users who rated the same items. In other word, users who rated a certain item contribute to rating prediction for the item. We introduce this relationship as the implicit influence based on information propagation.

Linear superposition: From the perspective of information propagation, user influence is regarded as the contribution for the diffusion of a message. Now, we investigate the global effect of user influence during the propagation process by the mean-field approximation.

The global effect is obtained from the global proportion of activation, i.e., the proportion of active users. Note that the global proportion of activation varies with time, so we define the proportion at time t as $\rho_a(t)$. Therefore, the proportion of inactive users is defined as $\rho_i(t)=1-\rho_a(t)$. In the mean-field approach, if the distribution of $P_{u,v}$ is given by $P(u,v)$, the average probability of being activated by an active neighbor is approximated by $\varepsilon=\sum_{u,v}P_{u,v}P(u,v)$.

Proposition 1: For the propagation of linear superposition, on a homogeneous network with node degree k, the global effect of user influence increases as an exponential function.

Proof: An inactive user is activated by one of his/her neighbors v in the interval Δt ($\Delta t\to 0$) with probability $\varepsilon\Delta t\delta_v$. According to Equation 4, an inactive user is activated during $[t,t+\Delta t]$ with the probability, as:

$$P_{i\to a}(t)=\sum_{v\in C_u}\varepsilon\Delta t\delta_v/|C_u|=\varepsilon\rho_a(t)\Delta t \tag{7}$$

The proportion of inactive users $\rho_i(t)$ decreases when inactive users are activated. Therefore, the variation of $\rho_i(t)$ during $[t,t+\Delta t]$ is $\rho_i(t)P_{i\to a}(t)$, and we obtain $\rho_i(t+\Delta t)=\rho_i(t)-\varepsilon\rho_i(t)\rho_a(t)\Delta t$. The transition rate of $\rho_i(t)$ can be calculated by:

$$\partial\rho_i(t)/\partial t=\frac{\lim_{\Delta t\to 0}\rho_i(t+\Delta t)-\rho_i(t)}{\Delta t}=-\varepsilon\rho_i(t)\rho_a(t) \tag{8}$$

The transition rate of $\rho_a(t)$ is obtained easily by $\partial\rho_a(t)/\partial t=\varepsilon\rho_i(t)\rho_a(t)$. From the two equations above, we obtain the time evolution of $\rho_a(t)$, shown as follows:

$$\rho_a(t)=\frac{1}{1+\mathrm{e}^{\ln(1/\rho_a(0)-1)-\varepsilon t}} \tag{9}$$

where $\rho_a(0)$ means the initial proportion of active users in the population.

Then, we consider information propagation on heterogeneous networks. In this case, the proportion of active users depends on users' degrees. The distribution of users' degrees is given by $P(k)$, and the probability that a user with degree k connects to a user with degree $k^{'}$ is given by $P(k^{'}|k)$. We define the proportions of active and inactive users with degree k as $\rho_a(k,t)$ and $\rho_i(k,t)$, respectively.

Proposition 2: For the propagation of linear superposition on a heterogeneous network, the effect of user influence increases as an exponential function.

Proof: In this case, the activation probability of an inactive user with degree k in Equation 7 is changed to:

$$\begin{aligned}P_{i\to a}(k,t)=&\frac{\sum_{v\in C_u}\varepsilon\Delta t\sum_{k^{'}}P(k^{'}|k)\rho_a(k^{'},t)}{|C_u|}\\=&\varepsilon\sum_{k^{'}}P(k^{'}|k)\rho_a(k^{'},t)\Delta t\end{aligned} \tag{10}$$

We calculate the transition rate of $\rho_i(k,t)$ as:

$$\partial\rho_i(k,t)/\partial t=-\varepsilon\rho_i(k,t)\sum_{k^{'}}P(k^{'}|k)\rho_a(k^{'},t) \tag{11}$$

From Equation 11, we obtain the time evolution of $\rho_a(k,t)$ as follows:

$$\rho_a(k,t)=1-\rho_i(k,0)\mathrm{e}^{-\varepsilon\phi(t)} \tag{12}$$

where $\phi(t)=\sum_{k^{'}}P(k^{'}|k)\int_0^t\rho_a(k^{'},t^{'})dt^{'}$. We conclude that user influence increases exponentially (see Appendix A). According to the aforementioned analysis, we model the effect of use influence as an exponential form of latent user and item vectors in recommender systems.

In the implicit influence of linear superposition, a user's rating on a certain item j is influenced by the users who rated item j, and the multiple influence of users accumulates linearly. We define the effect of user influence as:

$$\frac{1}{1+\mathrm{e}^{-V_j^T \sum_{i\in\Gamma_j} y_i}} \tag{13}$$

where Γ_j is the set of users who rated item j. y_i is the vector that indicates the implicit influence of user i who rated item j. Original ratings are regarded as explicit influence, and implicit influence can also be exploited from users' actions to better represent their preferences. $V_j^T \sum y_i$ is the accumulative information that a user receives from the users who rated item j during the process of information propagation.

Users always have different experience, expertise and feelings towards different items, so user influence is correlated with items. User influence on a specific user may be different when the user rates different items. Therefore, in Equation 13, user influence in rating prediction is not exerted on any specific user but on the users who rate a specific item.

With the implicit user influence, the estimated rating is given by:

$$\hat{r}_{u,j} = b_u + b_j + \mu + V_j^T U_u + \alpha \frac{1}{1+\mathrm{e}^{-V_j^T \sum_{i\in\Gamma_j} y_i}} \tag{14}$$

where b_u represents the inherent bias of user u, independently of any item, and b_j is the inherent bias of item j. Both of them are updated during the iterations of training. μ is the global average rating, and α is a positive constant which controls the extent of implicit influence of the users who rated the same item j. $V_j^T U_u$ is the inner product of latent user-specific and item-specific vectors, which represents explicit influence of the item.

Independent cascade: When a user who did not act is making a decision, he/she receives multiple information from active users, and multiple information generates cascade influence.

Proposition 3: For the propagation of independent cascade, on a homogeneous network with node degree k, the global effect of user influence increases as an exponential function.

The proof is analogous to Proposition 1. For an inactive user, the probability that he/she meets an active neighbor is $\rho_a(t)$. The activation probability $P_{i\to a}(t)$ during $[t, t+\Delta t]$ in Equation 7 is changed as:

$$P_{i\to a}(t) = 1 - \prod_{v\in C_u} (1-\varepsilon\Delta t \rho_a(t)) = 1-(1-\varepsilon\Delta t\rho_a(t))^k \tag{15}$$

Since Δt is very small, $P_{i\to a}(t)$ can be approximated by $k\varepsilon\Delta t\rho_a(t)$. For independent cascade, user influence also increases exponentially (see Appendix B).

Proposition 4: For the propagation of independent cascade on a heterogeneous network, the effect of user influence increases as an exponential function.

The proof is analogous to Proposition 2. For independent cascade, users who rated item j have their cascade influence on a new user, and the influence of these users is independent of each other. According to the exponential pattern of user influence, we define the independent cascade influence of all the users who rated item j as follows:

$$1 - \prod_{i\in\Gamma_j} \left(1 - \frac{1}{1+\mathrm{e}^{-V_j^T y_i}}\right) \tag{16}$$

where Γ_j is the set of users who rated item j. A natural and straightforward way to combine the implicit cascade influence with explicit influence from MF is given by:

$$\hat{r}_{u,j} = b_u + b_j + \mu + V_j^T U_u + \alpha\left(1 - \prod_{i\in\Gamma_j}(1 - \frac{1}{1+\mathrm{e}^{-V_j^T y_i}})\right) \tag{17}$$

Similar to linear superposition, α is a positive constant which controls the extent of implicit influence, and y_i is the implicit influence vector of user i in Γ_j.

D. Explicit trust influence

On social media, users can label or add other users as their trusted friends, and then, a trust network is generated. Assume that there are m users on a trust network. For the network, we get an adjacency matrix $\boldsymbol{C} = [c_{u,v}]_{m\times m}$ to describe the structure of the network, where $c_{u,v} = 1$ means that user u specifies user v as a trustworthy friend and $c_{u,v} = 0$ indicates no trust relationship. Similar to the rating matrix in MF, the trust matrix also can be used in MF. We denote U_u and W_v as truster u's latent factor vector and trustee v's latent vector. Aiming to bridge the rating matrix $\boldsymbol{R}$ and trust matrix $\boldsymbol{C}$ together, we limit the truster in $\boldsymbol{C}$ and the active user in $\boldsymbol{R}$ to share the same latent vector U_u. Then, we can use U_u and W_v to recover the trust matrix by:

$$\mathcal{L} = \frac{1}{2}\sum_u \sum_{v\in C_u^+} (\hat{c}_{u,v} - c_{u,v})^2 + \frac{\lambda}{2}\left(\sum_u ||U_u||^2 + \sum_v ||W_v||^2\right) \tag{18}$$

where $\hat{c}_{u,v}$ is the predicted trust relationship between user u and v by the inner product of truster and trustee vectors $\hat{c}_{u,v} = W_v^T U_u$, and C_u^+ is the set of users who are trusted by user u or the trustees of user u.

As explained before, we factorize the rating matrix and trust matrix together. With these two types of information, a new objective function is given by:

$$\mathcal{L} = \frac{1}{2}\sum_u \sum_{j\in I_u} (\hat{r}_{u,j} - r_{u,j})^2 + \frac{\lambda_t}{2}\sum_u \sum_{v\in C_u^+} (\hat{c}_{u,v} - c_{u,v})^2 \tag{19}$$

where λ_t controls the extent of trust regularization. In consideration of regularization terms, a unified objective function is proposed as follows:

$$\begin{aligned} \mathcal{L} = {} & \frac{1}{2}\sum_u \sum_{j\in I_u} (\hat{r}_{u,j} - r_{u,j})^2 + \frac{\lambda_t}{2}\sum_u \sum_{v\in C_u^+} (\hat{c}_{u,v} - c_{u,v})^2 \\ & + \frac{\lambda}{2}\Bigg(\sum_u ||U_u||^2 + \sum_j ||V_j||^2 + \sum_v ||W_v||^2 + \sum_i ||y_i||^2 \\ & \qquad + b_u^2 + b_j^2\Bigg) \end{aligned} \tag{20}$$

where $\hat{r}_{u,j}$ is defined in Equation 14 with linear superposition propagation or in Equation 17 with independent cascade propagation. $||*||$ is the Euclidean norm.

In summary, by making full use of the rating matrix and trust matrix, we blend the two kinds of information together, thereby making the prediction of $\hat{r}_{u,j}$ more appropriate. In addition, the implicit user influence based on information propagation is included in the prediction of $\hat{r}_{u,j}$.

E. Model learning

To optimize the objective function and determine a local minimum for Equation 20, we perform the following stochastic gradient descent (SGD) on different parameters across all the users and items in the training dataset. The rule of SGD is:

$$\theta = \theta - \eta * \nabla_\theta J(\theta) \tag{21}$$

In our paper, $J(\theta)$ is the objective function in Equation 20. Then the task of calculating $\nabla_\theta J(\theta)$ means to calculate $\nabla_{U_u}\mathcal{L}$, $\nabla_{V_j}\mathcal{L}$, $\nabla_{b_u}\mathcal{L}$, $\nabla_{b_j}\mathcal{L}$, $\nabla_{y_i}\mathcal{L}$, and $\nabla_{W_v}\mathcal{L}$ (see supplementary materials).

For linear superposition:

$$\begin{aligned}
U_u &\leftarrow U_u - \eta\Big(\sum_{j\in I_u} e_{u,j} * V_j + \lambda_t * \sum_{v\in C_u^+} s_{u,v} * W_v + \lambda * U_u\Big)\\
V_j &\leftarrow V_j - \eta\Big(\sum_{u\in\Gamma_j} e_{u,j} * (U_u + \alpha Y_j \frac{\mathrm{e}^{-V_j^T Y_j}}{(1+\mathrm{e}^{-V_j^T Y_j})^2}) + \lambda * V_j\Big)\\
b_u &\leftarrow b_u - \eta\Big(\sum_{j\in I_u} e_{u,j} + \lambda * b_u\Big)\\
b_j &\leftarrow b_j - \eta\Big(\sum_{u\in\Gamma_j} e_{u,j} + \lambda * b_j\Big)\\
y_i &\leftarrow y_i - \eta\Big(\alpha * \sum_{j\in I_i, u\in\Gamma_j} e_{u,j} * V_j * \frac{\mathrm{e}^{-V_j^T Y_j}}{(1+\mathrm{e}^{-V_j^T Y_j})^2} + \lambda * y_j\Big)\\
W_v &\leftarrow W_v - \eta\Big(\lambda_t * \sum_{u\in C_v^-} s_{u,v} * U_u + \lambda * W_v\Big)
\end{aligned} \tag{22}$$

where $Y_j = \sum_{i\in\Gamma_j} y_i$ is an auxiliary variable. $e_{u,j} = \hat{r}_{u,j} - r_{u,j}$ is the error of the estimated rating for user u on item j, and $s_{u,v} = \hat{c}_{u,v} - c_{u,v}$ is the error of estimated trust for user u to trustee v. C_u^+ is the set of users who are trusted by user u, and C_u^- is the set of users who trust user u.

For independent cascade, the update equations of U_u, b_u, b_j, and W_v are the same as those for linear superposition. We only list the updates of V_j and y_i for independent cascade as follows:

$$\begin{aligned}
V_j &\leftarrow V_j - \eta\Big(\sum_{u\in\Gamma_j} e_{u,j} * (U_u + \alpha D_j) + \lambda * V_j\Big)\\
y_i &\leftarrow y_i - \eta\Big(\alpha * \sum_{j\in I_i, u\in\Gamma_j} e_{u,j} * \frac{V_j}{1+\mathrm{e}^{-V_j^T y_i}} * Q_j + \lambda * y_i\Big)
\end{aligned} \tag{23}$$

where the auxiliary variables are $Q_j = \prod_{i\in\Gamma_j}\Big(1 - \frac{1}{1+\mathrm{e}^{-V_j^T y_i}}\Big)$ and $D_j = \sum_{i\in\Gamma_j}\Big(\frac{y_i}{1+\mathrm{e}^{-V_j^T y_i}} * Q_j\Big)$. The whole training algorithm is shown in Algorithm 1. Several arguments are taken as input, including the rating matrix $\boldsymbol{R}$, trust matrix $\boldsymbol{C}$, learning rate η, regularization parameter λ, trust regularization λ_t and user influence parameter α.

F. Complexity analysis

The computation mainly is caused by calculating the objective function in Equation 20 and its gradients versus different variables. We have defined m as the number of users, n as the number of items, and d as the number of latent factors. The average number of ratings received by an item is given by M. $|\boldsymbol{R}|$ represents the number of observed ratings, and $|\boldsymbol{C}|$ denotes the number of observed trust relationships.

For recommendation with linear superposition, during an iteration, it takes $O(dnM)$ to calculate Y_j for all the items. The computational complexities for the update of U_u, V_j and y_i are $O(d|\boldsymbol{R}|+d|\boldsymbol{C}|)$, $O(d|\boldsymbol{R}|)$, and $O(d|\boldsymbol{R}|M)$, respectively. Therefore, the complexity for linear superposition in an iteration is $O(d|\boldsymbol{R}|M)$. For independent cascade, the complexities of calculating Q_j and D_j are $O(dnM)$. The update of y_i during an iteration also costs $O(d|\boldsymbol{R}|M)$. Since $n \ll |\boldsymbol{R}|$, the computational complexity for independent cascade is the same as that for linear superposition.

The complexity of TrustSVD in an iteration is $O(d|\boldsymbol{R}|l + d|\boldsymbol{C}|l)$, where $l = \max(M, k^+, k^-)$. k^+ and k^- are the average number of trust statements given and received, respectively, by a user [28]. From the above analysis, the proposed recommendation method results in lower computational complexity than TrustSVD, and the computational time is linear with respect to the number of observed ratings. The complexity of TrustANLF [33] in an iteration is $O(d^2|\boldsymbol{R}|)$. The average number of ratings M received by an item in our datasets is less than 40, so M is close to d which in our experiments is set to 20. Therefore, the complexity of the proposed method approaches TrustANLF, and our method is scalable for practical use.

IV. Evaluation

In this section, we evaluate the effectiveness of the proposed method by extensive experiments, and we compare the results with those of other state-of-the-art recommendation models. We also discuss our findings in detail. All the experiments are performed on a PC with an Intel 2.9GHz CPU and 16GB RAM.

A. Datasets

Our datasets were collected from Ciao and Epinions, which are well-known and publicly available datasets[1] for social recommendation systems. Ciao[2] is a popular online website that works by critically reviewing and rating millions of

[1] https://www.cse.msu.edu/~tangjili/trust.html
[2] http://www.ciao.co.uk

Algorithm 1: Learning parameters in SoInp

Input: Rating matrix $\boldsymbol{R}$, trust matrix $\boldsymbol{C}$, λ, λ_t, η, α
Output: User factor matrix $\boldsymbol{U}$ and item factor matrix $\boldsymbol{V}$

Randomly initialize each U_u, V_j and y_i with small values in (0,0.01)
while *not convergence* **do**
  for *each item j in the item set* **do**
    if linear superposition: calculate $Y_j = \sum_{i \in \Gamma_j} y_i$
    if independent cascade: calculate $Q_j = \prod_{i \in \Gamma_j} \left(1 - \frac{1}{1+\mathrm{e}^{-V_j^T y_i}}\right)$ and $D_j = \sum_{i \in \Gamma_j} \left(\frac{y_i}{1+\mathrm{e}^{-V_j^T y_i}} * Q_j\right)$
  end
  calculate $\hat{r}_{u,j}$ using Equation 14 for linear superposition
    using Equation 17 for independent cascade
  calculate $e_{u,j} = \hat{r}_{u,j} - r_{u,j}$
  calculate $s_{u,v} = \hat{c}_{u,v} - c_{u,v}$
  Update b_u, b_j, U_u, W_v according to Equation 22
  if linear superposition: update V_j, y_i using Equation 22
  if independent cascade: update V_j, y_i using Equation 23
end

TABLE III
STATISTICS OF THE CIAO AND EPINIONS DATASETS

	Ciao	Epinions
Users	7267	7411
Items	11,211	8,728
Ratings	149,147	276,116
Trust Relationships	110,755	52,982

products for the benefit of the global population, and Epinions[3] is a well-known product review website established in 1999. In both websites, users can rate products by one of five discrete ratings from 1-5. Users observe ratings about a variety of items to help them decide on their actions. When a user is rating an item, he/she is influenced by other customers' ratings.

The Ciao dataset contains 149,147 ratings from 7,267 users on 11,211 items, and the Epinions dataset contains 276,116 ratings of 8,728 items made by 7,411 users. Users in Epinions and Ciao can specify others as their trustworthy friends by evaluating the quality of others' ratings and textual reviews. Therefore, in these two datasets we also obtain trust information which makes Ciao and Epinions become ideal sources for social recommendation experiments. Table III shows the statistics of these two datasets.

B. Experimental settings

We use two well-known metrics to evaluate the quality of recommendation, i.e., Mean Absolute Error (MAE) and Root Mean Square Error (RMSE), defined by:

$$MAE = \frac{\sum_{r_{u,j} \in R_{test}} |\hat{r}_{u,j} - r_{u,j}|}{|R_{test}|} \tag{24}$$

$$RMSE = \sqrt{\frac{\sum_{r_{u,j} \in R_{test}} (\hat{r}_{u,j} - r_{u,j})^2}{|R_{test}|}} \tag{25}$$

[3]http://www.epinions.com

where R_{test} is the test set, and $|R_{test}|$ is the number of ratings in the test set. From the definitions, we can see that a smaller MAE or RMSE value means better performance.

We apply several representative recommendation models on these two datasets, and compare their results with our method. The models include:

UserAverage: This baseline method predicts a user's rating on any item by the average of his/her historical ratings.

ItemAverage: This baseline method uses the mean value of ratings on each item to predict missing ratings.

NMF [39]: This method only uses the user-item rating matrix for recommendation with the assumption that factorized low-rank matrices have no negative element.

PMF [8]: This method is a basic probabilistic MF model, and it only uses the rating matrix.

RSTE [40]: This method makes trust-based recommendation that models one user's rating as the balance between his/her own favor and the tastes of his/her trusted users.

SocialMF [41]: This method makes social recommendation for a user based on ratings of the users that have direct or indirect social relationships with the target user.

TrustMF [34]: This method works to improve the performance of recommendation by integrating sparse rating data and sparse trust relationship data.

SVD++ [42]: This method is a state-of-the-art model based only on ratings, in which the implicit influence of items already rated by a user contributes to the prediction of missing ratings.

TrustSVD [28]: This method extends SVD++, and further incorporates both the explicit and implicit influence of trusted users on the recommendation.

TrustANLF [33]: This method incorporates users' trust information in a nonnegative MF framework.

For the propagation of linear superposition, our method is simplified as SoInp-LS, while for the propagation of independent cascade, it is written as SoInp-IC.

Parameter settings: We randomly select 80% of data in each dataset as the training set, and the remaining data are used as the test set. All results are averaged over five independent experiments. We implement some baselines by LibRec. To make a fair comparison, we set the same number of latent factors $d = 20$ for all the methods. We use cross-validation to determine the optimal parameters. All experiments are conducted under optimal settings for each model. In the following, the common setting of state-of-the-art models is the learning rate $\eta = 0.01$. The regularization parameter is $\lambda = 0.1$ for all the methods except TrustMF, TrustSVD and TrustANLF. The other settings are listed as follows:

1) RSTE: $\lambda_t = 1.8$ in Ciao, and $\lambda_t = 1$ in Epinions;
2) SocialMF: $\lambda_t = 1$ in Ciao, and $\lambda_t = 0.4$ in Epinions;
3) TrustMF: $\lambda_t = 2$, $\lambda = 0.05$ in Ciao, and $\lambda_t = 3$, $\lambda = 0.05$ in Epinions;
4) TrustSVD: $\lambda_t = 1$, $\lambda = 0.5$ in Ciao, and $\lambda_t = 0.5$, $\lambda = 0.5$ in Epinions;
5) TrustANLF: $\lambda_t = 1$, $\lambda = 0.5$ in Ciao, and $\lambda_t = 1$, $\lambda = 0.5$ in Epinions;
6) SoInp-LS: $\lambda_t = 5$, $\alpha = 1$, $\eta = 0.001$ in Ciao, and $\lambda_t = 4$, $\alpha = 1$, $\eta = 0.001$ in Epinions;
7) SoInp-IC: $\lambda_t = 5$, $\alpha = 0.8$, $\eta = 0.001$ in Ciao, and $\lambda_t = 4$, $\alpha = 1$, $\eta = 0.001$ in Epinions.

C. Impact of parameters

We investigate the impacts of parameters α, λ_t and d. These parameters have a significant role in our method.

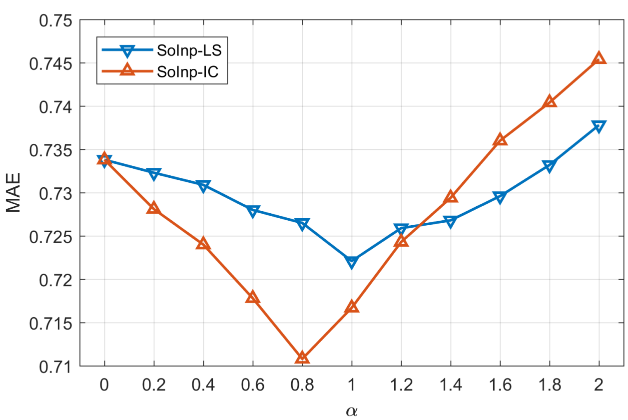


Fig. 2. Effect of parameter α in Ciao.

The parameter α determines the extent of implicit influence of users who rated the target item before. Figure 2 and Figure 3 show that, in both datasets, as α increases, the values of MAE for our method initially decrease, and then increase beyond a certain point of α. For very small α, only explicit user-item ratings dominate in the prediction task, and the implicit user influence fails to take effect in recommendation. However, for very large α, the implicit user influence has an excessive role in the training process and leads to over fitting. Increasing α makes the performance even worse. In addition, SoInp-LS in the Ciao dataset outperforms SoInp-IC for large α, while the results are contrary in other cases. The reason is that the rating matrix in Ciao is sparser, so large α promotes

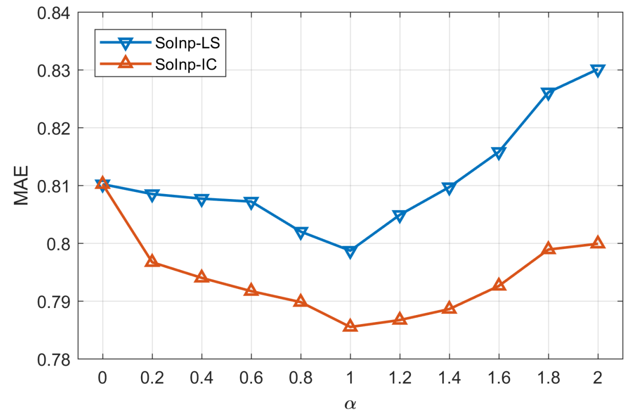


Fig. 3. Effect of parameter α in Epinions.

over fitting more easily in the dataset. From Eqs. (13) and (16), the value of independent cascade usually is larger than that of linear superposition. Therefore, the implicit user influence dominates in the rating prediction of SoInp-IC, resulting in relatively worse performance. To determine the optimal α for different datasets, we fix parameter λ_t at the best value, and tune α in the range (0.2,2). As depicted in the figures, the results show that our method achieves relatively low MAE in the range (0.8,1) for SoInp-LS and SoInp-IC in both datasets. The complexity of our method can be reduced, since it is convenient to set the parameter in different datasets.

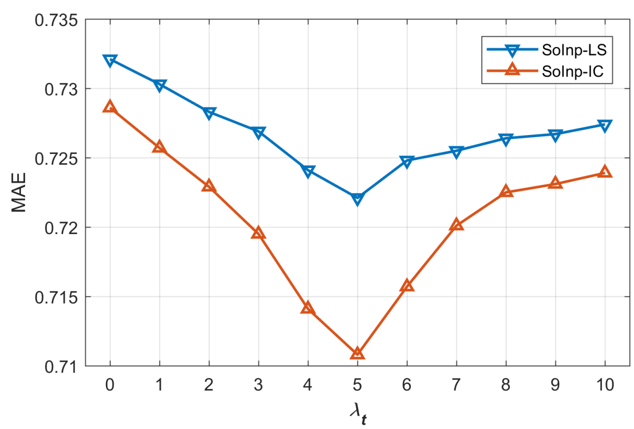


Fig. 4. Effect of parameter λ_t in Ciao.

The trust regularization parameter λ_t determines the extent of explicit social influence. Figure 4 and Figure 5 illustrate the impact of λ_t on MAE in our method. From the results, incorporating social information improves the performance of recommender systems. Similarly to α, if we use very small λ_t, we only leverage the rating matrix for the prediction of missing ratings, and social information is ignored. However, if we use very large λ_t, the contribution of observed ratings is restricted and the accuracy also is degraded. It is dramatic that in Ciao data, our method with large λ_t performs better than it does with small λ_t, but the result is just opposite in Epinions data. Compared with Ciao, Epinions data have sparser trust relationships, and therefore, too large λ_t harms the performance more prominently in Epinions. In addition,

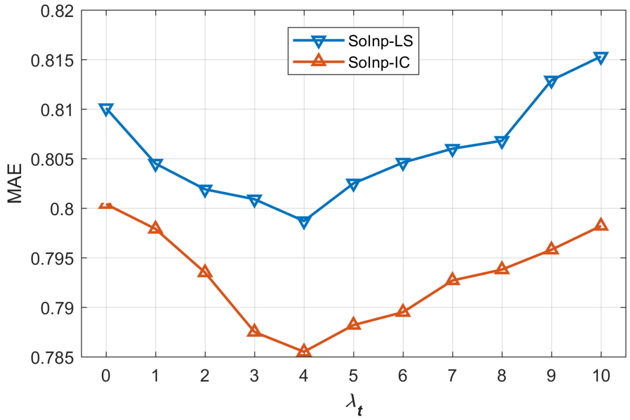


Fig. 5. Effect of parameter λ_t in Epinions.

we find the suitable value of λ_t is in the range (4,5) both for SoInp-LS and SoInp-IC, irrespective of the dataset.

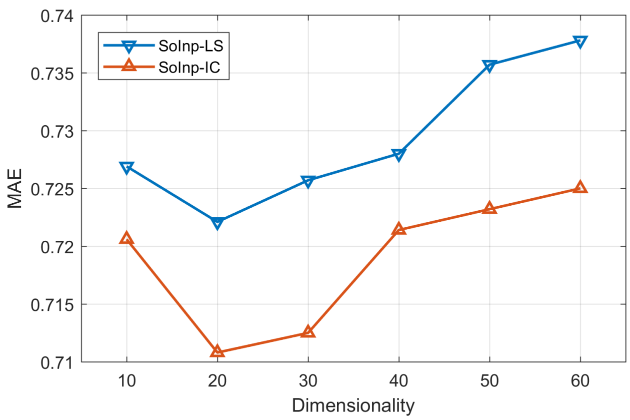


Fig. 6. Effect of dimensionality d in Ciao.

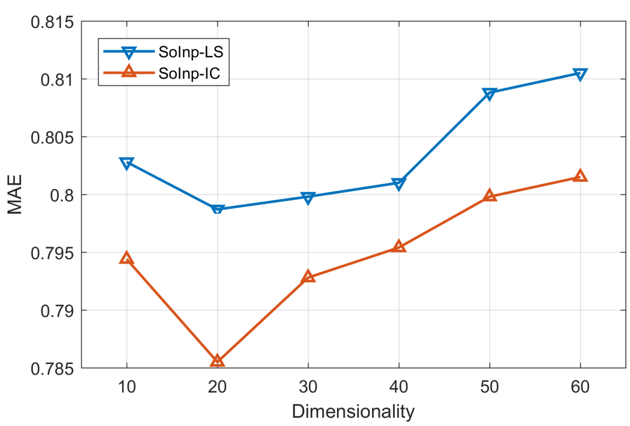


Fig. 7. Effect of dimensionality d in Epinions.

Here, we investigate the effect of the dimensionality of latent vectors d. Figure 6 and Figure 7 show the variations of MAE versus the dimensionality. It is observed that the value 20 of d is always optimal both for SoInp-LS and SoInp-IC in the two datasets. In fact, when the dimensionality surpasses a certain value, the model becomes overly complicated, and large dimensionality turns out to exert a negative impact on the prediction accuracy.

TABLE IV
PERFORMANCE COMPARISON OF SOINP-LS AND SOINP-IC

	Ciao		Epinions	
	MAE	RMSE	MAE	RMSE
SoInp-LS	0.7221	0.9703	0.7987	1.0411
SoInp-IC	**0.7108**	**0.9680**	**0.7855**	**1.0350**

TABLE V
PERFORMANCE COMPARISON ON MAE AND RMSE OVER DIFFERENT RECOMMENDATION MODELS

	Ciao		Epinions	
	MAE	RMSE	MAE	RMSE
UserAverage	0.8325	1.0993	0.9593	1.2119
ItemAverage	0.8588	1.1045	0.9302	1.1671
NMF	0.7844	1.0799	0.8725	1.1693
PMF	0.7665	1.0092	0.8734	1.1340
RSTE	0.7973	1.8994	0.8751	1.0951
SocialMF	0.7663	0.9829	0.8283	1.0567
TrustMF	0.7750	1.0976	0.8476	1.1149
SVD++	0.7375	0,9765	0.8054	1.0471
TrustSVD	0.7334	0.9710	0.8076	1.0430
TrustANLF	0.7285	0.9702	0.8068	1.0417
SoInp-IC	**0.7108**	**0.9680**	**0.7855**	**1.0350**

D. Comparison for normal users

We compare the accuracy of SoInp-LS and SoInp-IC in both datasets. The results are presented in Table IV. Here, we use optimal parameter settings. We conclude that the method with cascade information propagation consistently achieves better performance. Therefore, it is reasonable to assume that the influence of users who rated the same items is independent of each other, and the overall outcome is a cascade of different user influence. Independent cascade influence exists more extensively in real situations where users and their friends interact with each other and have an impact on others' decisions.

To demonstrate the performance improvement of our method, we compare it to representative state-of-the-art recommendation models. Parameters are assigned as mentioned earlier. The random split of data is conducted 5 times and the average experimental results are presented in Table V.

In fact, even small improvements in MAE and RMSE produce much better recommendation in practice. The baseline methods such as UserAverage and ItemAverage perform the worst among all the methods. Therefore, when the dataset is really sparse, these methods cannot be used for the task of personalized recommendation. In general, recommendation models with social information except RSTE have better performance than the models which depend only on ratings. RSTE utilizes social trust ensemble and requires more relationship data. Therefore, it cannot perform well with a sparse user-user trust matrix, especially for the Ciao dataset.

Furthermore, SVD++, despite social information, outperforms RSTE, SocialMF and TrustMF both in Ciao and Epinions. This result implies that the implicit influence of rated items identifies the actual interaction process on social media and makes adequate use of observed ratings. Therefore, the implicit influence can help improve the recommendation. TrustSVD is superior to SVD++ across both datasets, indicating that recommender systems can make further progress by incorporating the explicit and implicit influence of trust relationships with ratings. TrustANLF performs the best among these state-of-the-art models. In addition, our method outperforms other recommendation models in both datasets, demonstrating the effectiveness of leveraging the mechanism of information propagation to represent the implicit use influence. SoInp-IC in Ciao data decreases MAE by 3.08% in contrast to TrustSVD, and 2.74% in Epinions data. In addition, SoInp-IC has lower computational complexity than TrustSVD.

To illustrate the importance of information propagation, we present the Pearson correlation coefficient (PCC) between a user's ratings on different items and implicit user influence towards these items in Equation 13 and Equation 16, as shown in Figure 8 and Figure 9. A large portion (about 65% in Ciao and 77% in Epinions) of the correlations are above 0, implying that a user's ratings are positively correlated with implicit user influence both for SoInp-LS and SoInp-IC. The distributions for SoInp-LS and SoInp-IC in both datasets are approximately the same. Therefore, it is obvious that implicit user influence from information propagation can be utilized for rating prediction.

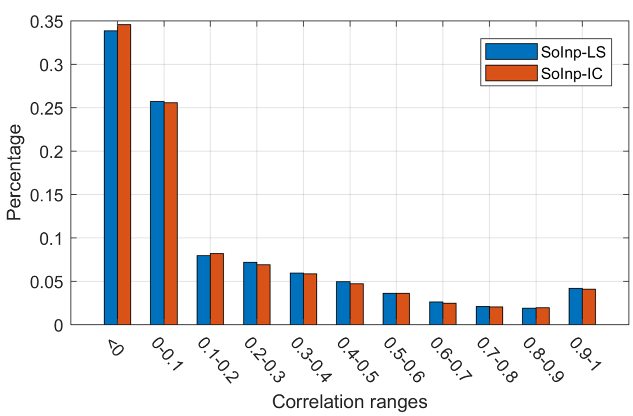


Fig. 8. Correlations between a user's ratings on items and implicit user influence towards these items in Ciao.

The experimental results suggest that the implicit user influence based on information propagation improves the recommendation accuracy. Further experiments can be found in supplementary materials.

E. Time analysis

In this section, the recommendation models are compared in terms of time consuming, and the results are shown in Figure 10 for Ciao and Epinions, respectively. We select 6 most competitive recommendation models that perform well

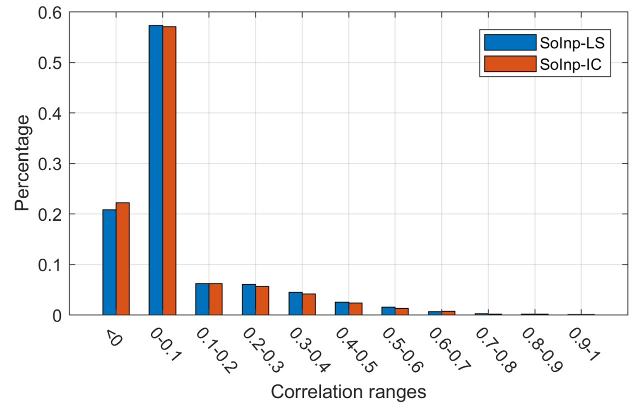


Fig. 9. Correlations between a user's ratings on items and implicit user influence towards these items in Epinions.

in experiments. As shown in Figure 10, SoInp-LS and SoInp-IC both require less execution time than TrustSVD while they achieve lower MAE and RMSE (in Table V). Although the proposed method spends more time than SocialMF, SVD++ and TrustANLF, the accuracy of our method is much better than these models. Therefore, the additional execution time is worth achieving better recommendation performance. In addition, our method is also sufficiently scalable (see supplementary materials).

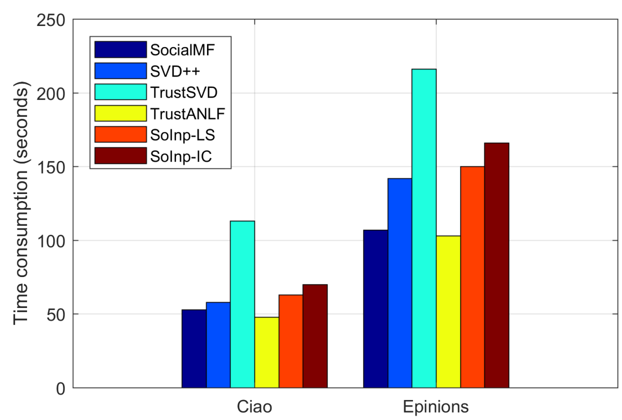


Fig. 10. Execution time of different models.

F. Comparison for cold-start users

In this subsection, we analyze the ability of addressing cold-start problems for our method. Users who have given few ratings are considered as cold-start users. In both datasets, we choose the users who have rated less than 10 items as cold-start users. Table VI shows the MAE and RMSE performance of different models for cold-start users in both datasets.

In cold-start situations, our method performs the best among these recommendation models. RSTE for cold-start users outperforms PMF, but the result is opposite to that for all users. TrustMF even has slightly smaller MAE than SVD++ in both datasets. The reason is that trust information becomes more important in rating prediction, when observed ratings are not enough to train a predictive model in cold-start situations. In

TABLE VI
PERFORMANCE COMPARISON FOR COLD-START USERS OVER DIFFERENT RECOMMENDATION MODELS

	Ciao		Epinions	
	MAE	RMSE	MAE	RMSE
UserAverage	0.8479	1.1521	1.0067	1.3147
ItemAverage	0.8409	1.1680	0.9880	1.2370
NMF	0.8167	1.1494	0.9079	1.2412
PMF	0.7993	1.0954	0.9606	1.2103
RSTE	0.7677	1.1411	0.9500	1.1731
SocialMF	0.7779	0.9862	0.9003	1.1439
TrustMF	0.7635	1.0501	0.8933	1.1991
SVD++	0.7676	0.9883	0.8995	1.1405
TrustSVD	0.7410	0.9752	0.8662	1.1299
TrustANLF	0.7398	0.9689	0.8388	1.1009
SoInp-IC	**0.7152**	**0.9612**	**0.7886**	**1.0431**

addition, since trust relationships are much sparser in Epinions than in Ciao, our method performs much better than other state-of-the-art recommendation models in Epinions. In terms of MAE, SoInp-IC in Ciao improves the accuracy by 3.48% in contrast to TrustSVD, and by 8.96% in Epinions. Therefore, our method efficiently leverages trust information when rating data are sparse.

V. CONCLUSION

User connections on online social networks are different and diverse. Users seem to interact more frequently with others they trust. In addition, users' ratings also are influenced by the users who rated the same items even if there is no direct connection among them. In this paper, we focused on the measure of implicit influence propagation during users' interactions, which often was overlooked by existing recommender systems. We studied the effect of implicit user influence based on information propagation, and we concluded the concrete function of implicit influence by theoretical analysis. Then, we combined the implicit influence and explicit trust information with the MF framework. Our experiments on two datasets indicate that our method achieves more accurate recommendation than state-of-the-art models. Furthermore, the proposed method is quite general, since it also can be applied even when social networks are not available.

There are several ways to improve the recommendation in the future. In our current work, we just simply take into account all the users who rated the same items. In fact, a user who has different similarities with others may have different influence on those users, so we will investigate how to take advantage of user similarities and the implicit influence for recommendation. Meanwhile, users' social relationships change over time, and trust networks are likely to evolve since new members may join and old members may leave. Therefore, we will seek to use temporal information of both ratings and social relationships to improve the recommendation.

ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China under Grant No. 61872033, the Humanity and Social Science Youth Foundation of Ministry of Education of China under Grant No. 18YJCZH204, and the Beijing Natural Science Foundation under Grant No. 4184084. (Corresponding author: Hongshu Chen.)

APPENDIX A

In an uncorrelated network, the degree correlation of the network can be written as $P(k'|k) = k'P(k')/\bar{k}$, where $\bar{k}$ is the average degree of the network. We multiply both sides of Equation 12 with $kP(k)/\bar{k}$ and sum over different k, as:

$$\partial\phi(t)/\partial t = \sum_k kP(k)(1 - \rho_i(k,0)e^{-\varepsilon\phi(t)})/\bar{k} \tag{26}$$

We calculate the partial derivative of Equation 26, and obtain the following equation:

$$\partial^2\phi(t)/\partial t^2 = \partial\phi(t)/\partial t \sum_k \varepsilon e^{-\varepsilon\phi(t)}\rho_i(k,0)kP(k)/\bar{k} \tag{27}$$

Considering Equation 26 and Equation 27, we have:

$$\partial^2\phi(t)/\partial t^2 = \varepsilon\partial\phi(t)/\partial t(1 - \partial\phi(t)/\partial t) \tag{28}$$

We solve the above equation as:

$$\partial\phi(t)/\partial t = \frac{1}{1 + e^{ln(1/\partial_t\phi(0)-1)-\varepsilon t}} \tag{29}$$

where $\partial_t\phi(0) = \sum_k kP(k)\rho_a(k,0)/\bar{k}$. According to Equation 4, an inactive user with degree k is influenced by others with the following probability:

$$\varepsilon\sum_{k'} P(k^{'}|k)\rho_a(k^{'},t) = \varepsilon\frac{\partial\phi(t)}{\partial t} = \frac{\varepsilon}{1 + e^{ln(1/\partial_t\phi(0)-1)-\varepsilon t}} \tag{30}$$

If initial active users are uniformly distributed over different degrees, the time evolution of $\rho_a(k,t)$ is reduced to Equation 9. According to above analysis, user influence increases exponentially.

APPENDIX B

In Proposition 3, analogously to the case of linear superposition, we can obtain the evolution of $\rho_a(t)$ for independent cascade:

$$\rho_a(t) = \frac{1}{1 + e^{ln(1/\rho_a(0)-1)-k\varepsilon t}} \tag{31}$$

According to Equation 5, each user is influenced by others with the following probability:

$$1-\prod_k(1-\varepsilon\rho_a(t)) = 1-\prod_k(1-\frac{\varepsilon}{1 + e^{ln(1/\rho_a(0)-1)-k\varepsilon t}}) \tag{32}$$

Therefore, for the propagation of independent cascade, user influence increases exponentially.

References

[1] S. Deng, D. Wang, X. Li, and G. Xu, "Exploring user emotion in microblogs for music recommendation," *Expert Systems with Applications*, vol. 42, no. 23, pp. 9284–9293, 2015.

[2] J. Golbeck, "Generating predictive movie recommendations from trust in social networks," in *Proceedings of the 4th International Conference on Trust Management*, 2006, pp. 93–104.

[3] M. Y. Hsieh, W. K. Chou, and K. C. Li, "Building a mobile movie recommendation service by user rating and app usage with linked data on hadoop," *Multimedia Tools and Applications*, vol. 76, no. 3, pp. 1–19, 2016.

[4] F. Alqadah, C. K. Reddy, J. Hu, and H. F. Alqadah, "Biclustering neighborhood-based collaborative filtering method for top-n recommender systems," *Knowledge and Information Systems*, vol. 44, no. 2, pp. 475–491, 2015.

[5] R. Jin, J. Y. Chai, and L. Si, "An automatic weighting scheme for collaborative filtering," in *Proceedings of the 27th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2004, pp. 337–344.

[6] M. Deshpande and G. Karypis, "Item-based top-n recommendation algorithms," *ACM Transactions on Information Systems*, vol. 22, no. 1, pp. 143–177, 2004.

[7] M. Y. Hsieh, T. H. Weng, and K. C. Li, "A keyword-aware recommender system using implicit feedback on hadoop," *Journal of Parallel and Distributed Computing*, p. S0743731517303398, 2018.

[8] R. Salakhutdinov and A. Mnih, "Probabilistic matrix factorization," in *Proceedings of the 21st International Conference on Neural Information Processing Systems*, 2007, pp. 1257–1264.

[9] B. Li, X. Zhu, R. Li, and C. Zhang, "Rating knowledge sharing in cross-domain collaborative filtering," *IEEE Transactions on Cybernetics*, vol. 45, no. 5, pp. 1068–1082, 2015.

[10] S. Pyo, E. Kim, and M. Kim, "Lda-based unified topic modeling for similar tv user grouping and tv program recommendation." *IEEE Transactions on Cybernetics*, vol. 45, no. 8, pp. 1476–1490, 2015.

[11] X. Wang, Y. Liu, J. Lu, F. Xiong, and G. Zhang, "Trugrc: Trust-aware group recommendation with virtual coordinators," *Future Generation Computer Systems*, vol. 94, pp. 224–236, 2019.

[12] G. Guo, J. Zhang, and D. Thalmann, "A simple but effective method to incorporate trusted neighbors in recommender systems," in *International Conference on User Modeling, Adaptation, and Personalization*, 2012, pp. 114–125.

[13] P. Hao, G. Zhang, L. Martinez, and J. Lu, "Regularizing knowledge transfer in recommendation with tag-inferred correlation," *IEEE Transactions on Cybernetics*, no. 99, pp. 1–14, 2017.

[14] H. Ma, H. Yang, M. R. Lyu, and I. King, "Sorec: social recommendation using probabilistic matrix factorization," in *Proceedings of the 17th ACM conference on Information and Knowledge Management*, 2008, pp. 931–940.

[15] H. Ma, D. Zhou, C. Liu, M. R. Lyu, and I. King, "Recommender systems with social regularization," in *Proceedings of the 4th ACM International Conference on Web Search and Data Mining*, 2011, pp. 287–296.

[16] M. Jamali and M. Ester, "A matrix factorization technique with trust propagation for recommendation in social networks," in *Proceedings of the 4th ACM Conference on Recommender Systems*, 2010, pp. 135–142.

[17] B. Yang, Y. Lei, J. Liu, and W. Li, "Social collaborative filtering by trust," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 8, pp. 1633–1647, 2017.

[18] H. Fang, B. Yang, and J. Zhang, "Leveraging decomposed trust in probabilistic matrix factorization for effective recommendation," in *Proceedings of the 28th AAAI Conference on Artificial Intelligence*, vol. 350, 2014.

[19] S. Deng, L. Huang, and G. Xu, "Social network-based service recommendation with trust enhancement," *Expert Systems with Applications*, vol. 41, no. 18, pp. 8075–8084, 2014.

[20] A. Javari and M. Jalili, "A probabilistic model to resolve diversity–accuracy challenge of recommendation systems," *Knowledge and Information Systems*, vol. 44, no. 3, pp. 609–627, 2015.

[21] Q. Bao, W. K. Cheung, Y. Zhang, and J. Liu, "A component-based diffusion model with structural diversity for social networks." *IEEE Transactions on Cybernetics*, vol. 47, no. 4, pp. 1078–1089, 2017.

[22] N. Guan, D. Tao, Z. Luo, and B. Yuan, "Online nonnegative matrix factorization with robust stochastic approximation," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 7, pp. 1087–1099, 2012.

[23] H.-F. Yu, C.-J. Hsieh, S. Si, and I. S. Dhillon, "Parallel matrix factorization for recommender systems," *Knowledge and Information Systems*, vol. 41, no. 3, pp. 793–819, 2014.

[24] R. Kannan, M. Ishteva, and H. Park, "Bounded matrix factorization for recommender system," *Knowledge and Information Systems*, vol. 39, no. 3, pp. 491–511, 2014.

[25] M. Mao, J. Lu, G. Zhang, and J. Zhang, "Multirelational social recommendations via multigraph ranking," *IEEE Transactions on Cybernetics*, vol. 47, no. 12, pp. 4049–4061, 2017.

[26] W. Yao, J. He, G. Huang, and Y. Zhang, "Modeling dual role preferences for trust-aware recommendation," in *Proceedings of the 37th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2014, pp. 975–978.

[27] S. Tan, J. Bu, C. Chen, B. Xu, C. Wang, and X. He, "Using rich social media information for music recommendation via hypergraph model," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 7, no. 1, p. 22, 2011.

[28] G. Guo, J. Zhang, and N. Yorke-Smith, "A novel recommendation model regularized with user trust and item ratings," *IEEE Transactions on Knowledge and Data*

Engineering, vol. 28, no. 7, pp. 1607–1620, 2016.

[29] 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 *Proceedings of the 20th International Conference on World Wide Web*, 2011, pp. 537–546.

[30] P. Massa and P. Avesani, "Trust-aware recommender systems," in *Proceedings of the 2007 ACM Conference on Recommender Systems*, 2007, pp. 17–24.

[31] ——, *Trust-Aware Collaborative Filtering for Recommender Systems*. Springer Berlin Heidelberg, 2004.

[32] R. Guha, R. Kumar, P. Raghavan, and A. Tomkins, "Propagation of trust and distrust," in *Proceedings of the 13th International Conference on World Wide Web*, 2004, pp. 403–412.

[33] H. Parvin, P. Moradi, S. Esmaeili, and N. N. Qader, "A scalable and robust trust-based nonnegative matrix factorization recommender using the alternating direction method," *Knowledge-Based Systems*, vol. 166, pp. 92–107, 2019.

[34] B. Yang, Y. Lei, D. Liu, and J. Li, "Social collaborative filtering by trust," in *Proceedings of the 23rd International Joint Conference on Artificial Intelligence*, 2013, pp. 2747–2753.

[35] H. Parvin, P. Moradi, S. Esmaeili, and M. Jalili, "An efficient recommender system by integrating non-negative matrix factorization with trust and distrust relationships," in *The 2018 IEEE Data Science Workshop*. IEEE, 2018, pp. 135–139.

[36] D. Sun, T. Zhou, J.-G. Liu, R.-R. Liu, C.-X. Jia, and B.-H. Wang, "Information filtering based on transferring similarity," *Physical Review E*, vol. 80, p. 017101, 2009.

[37] M. Granovetter, "Threshold models of collective behavior," *American Journal of Sociology*, vol. 83, no. 6, pp. 1420–1443, 1978.

[38] J. Goldenberg, B. Libai, and E. Muller, "Talk of the network: A complex systems look at the underlying process of word-of-mouth," *Marketing Letters*, vol. 12, no. 3, pp. 211–223, 2001.

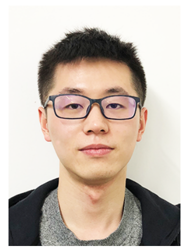
[39] D. D. Lee and S. H. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, p. 788, 1999.

[40] H. Ma, I. King, and M. R. Lyu, "Learning to recommend with social trust ensemble," in *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2009, pp. 203–210.

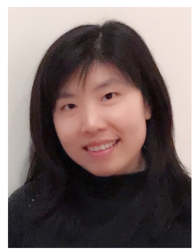
[41] M. Jamali and M. Ester, "A matrix factorization technique with trust propagation for recommendation in social networks," in *Proceedings of the 2010 ACM Conference on Recommender Systems*, 2010, pp. 135–142.

[42] Y. Koren, "Factorization meets the neighborhood: a multifaceted collaborative filtering model," in *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2008, pp. 426–434.

Fei Xiong received the B.E. degree and the Ph.D. degree in communication and information systems from Beijing Jiaotong University, Beijing, China, in 2007 and 2013. He is currently an Associate Professor with the School of Electronic and Information Engineering, Beijing Jiaotong University. From 2011 to 2012, he was a visiting scholar at Carnegie Mellon University. He has published over 60 papers in refereed journals and conference proceedings. He was a recipient of National Natural Science Foundations of China and several other research grants. His current research interests include the areas of web mining, complex networks and complex systems.

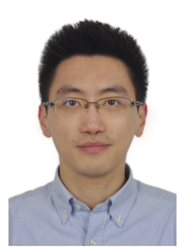


Weihan Shen received the B.E. degree in communication engineering from Beijing Jiaotong University, Beijing, China, in 2017. He is currently a Master student with School of Electronic and Information Engineering, Beijing Jiaotong University. His current research interests include social recommendation and information propagation.



Hongshu Chen received the dual Ph.D. degrees in Management Science and Engineering from Beijing Institute of Technology, Beijing, China, in 2015, and in Software Engineering from University of Technology Sydney, NSW, Australia, in 2016. She is currently an Assistant Professor with the School of Management and Economics, Beijing Institute of Technology. Her research interests include bibliometrics, information systems and text analytics.

Shirui Pan (M'16) received the Ph.D. degree in computer science from the University of Technology Sydney (UTS), NSW, Australia. He is currently a Lecturer with the Faculty of Information Technology, Monash University, VIC, Australia. Prior to that, he was a Research Fellow and then a Lecturer with the School of Software, University of Technology Sydney. His research interests include data mining and machine learning. To date, Dr. Pan has published over 50 research papers in top-tier journals and conferences, including the IEEE Transactions on Neural Networks and Learning Systems (TNNLS), IEEE Transactions on Knowledge and Data Engineering (TKDE), IEEE Transactions on Cybernetics (TCYB), ICDE, AAAI, IJCAI, and ICDM.



Ximeng Wang received the B.E. degree in communication engineering from the Nanjing University of Posts and Telecommunications, Nanjing, China, in 2011, and the M.E. degree in software engineering from Beijing Jiaotong University, Beijing, China, in 2013. He is currently pursuing the Ph.D. degree with Beijing Jiaotong University. Since 2017, he has been a joint Ph.D. student with the University of Technology Sydney. He has published over 10 papers in refereed journals and conference proceedings. His current research interests include recommender systems, complex networks and data mining.

Zheng Yan received the Ph.D. degree in mechanical and automation engineering from the Chinese University of Hong Kong, Hong Kong, in 2014. He is currently a Chancellor's Research Fellow in the Centre for Artificial Intelligence, Faculty of Engineering and Information Technology, University of Technology Sydney. Prior to this position, he took a leadership role in the research and development of deep learning systems in industry. He is an Area Editor for International Journal of Computational Intelligence Systems and a Guest Editor for Neurocomputing. His research interests include recurrent neural networks, optimization and embodied intelligence.