Adaptive Implicit Friends Identification over Heterogeneous Network for Social Recommendation

Junliang Yu[1,3], Min Gao[1,2], Jundong Li[2], ~~Hongzhi Yin~~[3], Huan Liu[2]
[1]Chongqing University, yu.jl@cqu.edu.cn
[2]Arizona State University, {mingao, jundongl, huan.liu} @asu.edu
[3]The University of Queensland, db.hongzhi@gmail.com

ABSTRACT

The explicitly observed social relations from online social platforms have been widely incorporated into conventional recommender systems to mitigate the data sparsity issue. However, the direct usage of explicit social relations may lead to an inferior performance due to the unreliability (e.g., noises) of observed links. To this end, the discovery of reliable relations among users plays a central role in advancing social recommender systems. In this paper, we propose a novel approach to adaptively identify implicit friends toward the discovery of more credible user relations. In particular, implicit friends are those who share similar tastes but could be distant from each other on the network topology of social relations. Methodologically, to find the implicit friends for each user, we first model the whole system as a heterogeneous information network, and then capture the similarity of users through embedding representation learning. Finally, our approach adaptively incorporates different amounts of similar users as implicit friends for each user to alleviate the adverse consequences of unreliable social relations for a more effective recommendation. Experimental analysis on three real-world datasets demonstrates the superiority of our method and explains why the implicit friends are helpful in improving the performance of social recommendation.

KEYWORDS

Social Recommender Systems, Implicit Friends, Heterogeneous Networks, Social Networks

ACM Reference Format:
Junliang Yu[1,3], Min Gao[1,2], Jundong Li[2], Hongzhi Yin[3], Huan Liu[2]. 2018. Adaptive Implicit Friends Identification over Heterogeneous Network for Social Recommendation. In *Proceedings of ACM CIKM conference (CIKM'18).* ACM, New York, NY, USA, 10 pages. https://doi.org/10.475/123_4

1 INTRODUCTION

The emergence and advancement of recommender systems have managed to mitigating the problem of *information overload*. However, in traditional recommender systems, most users usually only consume few of the millions of items, leading to an inferior recommendation accuracy because of the data sparsity problem [34]. Due to the explosive development of online social platforms, the explicitly observed social relations now can be harnessed to alleviate the data sparsity problem confronted by traditional recommender systems; for the reason that user preferences can be inferred from those of their friends [26, 33, 39]. With this intuition, social recommender systems [19] emerged and have attracted increasing attention over the past years. Nonetheless, recent studies reveal that social recommender systems suffer from the following issues: (1) explicit social relations are not always available in real-world recommender systems and are generally very sparse; (2) social relations may be very noisy due to the existence of spammers and bots. In addition to that, social relations have different interpretations in different contexts [30, 37]. For example, two close friends may reach a consensus on movies but have rather diverse opinions on purchasing clothes. Without the further filtering, the direct usage of explicit social relations may have an adverse impact on the recommendation quality.

A vast majority of existing social recommender systems [11, 14, 16, 40, 41, 46] based on matrix factorization [12] integrate the explicit social relations directly. Therefore, they are very likely to suffer from the limitations discussed above. Furthermore, the aforementioned approaches are fundamentally based on the assumption that connected users have similar tastes while unconnected users are more likely to have different preferences. But in reality users may also share similar tastes with other users that are distant from each other on the social network, and we refer to such user pairs as *implicit friends*. To this end, the nuanced approaches which can uncover and exploit reliable implicit social relations for recommendation should be studied. Despite the fact that the user-item bipartite network and user social network are two distinct networks with different types of nodes and connections, they are inherently correlated as users are involved on both networks. Hence, rich information is shared across these two networks and a wiser choice is to concatenate them as a whole heterogeneous information network (HIN) to perform further analysis. In this way, we are able to better capture the interactions among users in the system. For example, if two users purchased the same product while they are connected in the social network, their connection could be strengthened.

A few studies [35, 38, 42, 45] have explored heterogeneous information networks for general recommendation, but none of them paid attention to address the problem of unreliability of the explicit links for the social recommendation. In this paper, we propose a novel HIN based social recommendation method, which consists of two stages, to incorporate implicit friends to enhance the item recommendation. The implicit friends with maximum similarities for each user are identified in the first stage while the second stage focuses on effectively harnessing these implicit friends. The challenge in the first stage is how to uncover the potential implicit relations lying in the HIN. To handle it, we carefully design a set

CIKM'18, October 2018, Lingotto, Turan, Italy

ACM ISBN 123-4567-24-567/08/06.
https://doi.org/10.475/123_4

of meaningful meta-paths over the HIN which are composed of the user-item relations and the user-user social relations. These elaborately designed meta-paths enable us to identify Top-K users with the most similar tastes even though they are topologically distant from each other in the social networks. Concretely, the phase for implicit friends identification is performed by learning embedding representations for users in the network. In this way, we can alleviate the adverse effects resulted from unreliable explicit social relations. When the Top-K implicit friends for each user are identified, we then incorporate these implicit friends into an item ranking model. Then a natural question to ask is that do all users need the same amount of implicit friends to facilitate recommendation? Motivated by the findings that positive social effect is not observed in all groups of users in the explicit social network [33], we argue that integrating a fixed number (e.g. $k = 100$ for each user) of implicit friends for all users may result in a suboptimal solution. In consequence, we have developed a novel social BPR model to adaptively refine the Top-K implicit friends in a way that each user is able to select the optimal amount of friends according to their ranking performances on the observed feedbacks, which is superior to other methods based on implicit friends [29, 43]. In other words, the number of selected implicit friends is personalized rather than being globally fixed. To this end, an Expectation-Maximization (EM) algorithm based learning policy is employed to iteratively update the personalized smilarity threshold with regard to implicit friends for each user.

To summarize, our main contributions are listed as follows:

- We formally introduce the concept of implicit friends to social recommendation and show how these friends are accurately identified over the HIN by carefully designed meta-paths and embedding representation learning.
- We categorize user feedbacks into five sets and elaborately design a novel social BPR model which is capable of generating more effective recommendation.
- An EM algorithm based learning policy is adopted to adaptively refining the optimal Top-K implicit friends for each user, resulting in an obvious improvement of recommendation quality on different metrics.
- We rigorously conduct experiments to validate the effectiveness of the proposed approach in recommending items and explain why the implicit friends can improve the social recommendation.

The remainder of this paper is organized as follows. Section 2 introduces the preliminaries to facilitate the understanding of the developed approach. Section 3 illustrates how to identify Top-K implicit friends over HIN. Section 4 presents the proposed social recommender system in details. Experimental evaluations on real-world datasets are presented in section 5 with discussions. We briefly analyze the related work in Section 6. Finally, Section 7 concludes the whole paper.

2 PRELIMINARIES

In the task of social recommendation, we denote $\mathcal{U}$ as the user set and $\mathcal{I}$ as the item set. Also, we use $\mathbf{G_r}$ and $\mathbf{G_s}$ to denote the user-item bipartite network and the user-user social network, respectively. The network $\mathbf{G_r} = (V_r, E_r)$ contains two types of nodes, user and item, where $(u, i) \in E_r$ indicates that the user u purchased/rated the item i. The $\mathbf{G_s} = (V_s, E_s)$ only contains one type of node, which is the user, where $(u_1, u_2) \in E_s$ indicates that the user u_1 trusts the user u_2. The relationships between users are asymmetric and the edge (u_1, u_2) is different from the edge (u_2, u_1).

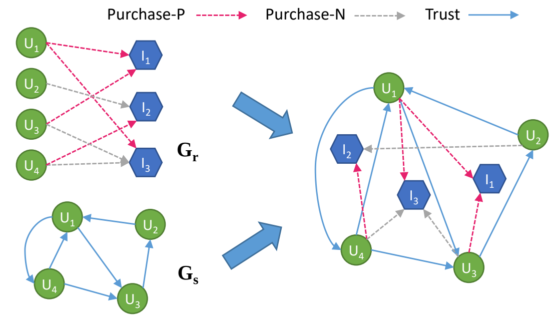

Figure 1: Heterogeneous information network constructed by the user-item bipartite network and the social network. Purchase-P denotes the consumption with positive feedbacks and Purchase-N denotes the consumption with negative feedbacks.

Motivated by the existing studies [42, 45], we consider that the promising way for social recommendation is to concatenate $\mathbf{G_r}$ and $\mathbf{G_s}$ as a whole to a *heterogeneous information network* [27] and study the recommendation problem based on the new network $\mathbf{H}$. In this regard, it enables us to capture rich information shared across $\mathbf{G_r}$ and $\mathbf{G_s}$ in quantifying user similarity for the social recommendation.

Definition 1. **Heterogeneous Information Network**: In a heterogeneous information network $\mathbf{H} = (V, E, T)$, each node v and each link e is associated with a mapping function $\phi(v) : V \longrightarrow T_V$ and $\phi(e) : E \longrightarrow T_E$, respectively. T_V and T_E denote the sets of object and relation types, where $|T_V| + |T_E| > 2$.

Figure 1 is an illustration of the HIN constructed by $\mathbf{G_r}$ and $\mathbf{G_s}$ where two types of nodes and three types of edges are involved. Different from the existing implicit feedback based recommender systems, in this work we make use of the negative feedbacks [32, 43]. As suggested by [8, 25], implicit negative feedbacks are common and valuable to examine. For example, out of curiosity, a user may click a song to listen, but soon she realizes that it is out of her taste and then closes it. In most of the popular recommender systems [22, 24], the above-mentioned clicks are treated as positive feedbacks, which is ill-considered. In our model, we also leverage the negative feedbacks and attempt to exploit them in a proper way.

Definition 2. **Implicit Friends**: Implicit friends refer to a pair of users with similar tastes or preferences but are not necessarily connected with each other on the social network.

This paper aims to identify implicit friends for each user in the social network and leverage the implicit friend relations to improve the performance of the social recommendation.

3 IDENTIFYING THE IMPLICIT FRIENDS OVER HIN

Identifying the implicit friends over HIN is the first important pillar of our developed approach. In this stage, the proposed method first

Table 1: Meta-paths designed for social recommendation.

Path	Schema	Description
P_1	$U \xrightarrow{p} I \xleftarrow{p} U$	Users who have consumed the same item are similar with each other
P_2	$U \xrightarrow{t} U \xrightarrow{t} U$	A user may trust her/his friends' friends
P_3	$U \xleftarrow{t} U \xrightarrow{t} U$	Users who are trusted by the same user are similar with each other
P_4	$U \xrightarrow{t} U \xleftarrow{t} U$	Users who share the same friends are similar with each other
P_5	$U \xrightarrow{t} U \xrightarrow{p} I \xleftarrow{p} U$	A user may have similar taste with someone who has similar taste with his/her trustee
P_6	$U \xrightarrow{t} U \xrightarrow{t} U \xrightarrow{p} I \xleftarrow{p} U$	Help find users of the similar tastes that are distant from each other on the social network

* $\xrightarrow{p}$ denotes the purchase relation and $\xrightarrow{t}$ denotes the trust relation.

generates biased meta-path based random walks to explore the HIN; and then learns node embedding representations via heterogeneous Skip-Gram to identify the Top-K implicit friends for each user.

3.1 Generating Social Corpora over HIN

Since the real-world recommender systems modeled by HINs are often in large-scale, containing millions of users and items, then the first challenge in identifying implicit friends is how to reduce the computational cost while preserving the information embedded in the original networks. Inspired by the success of network embedding models [4, 13, 23], we design a set of meaningful meta-paths over the HIN, and then conduct biased meta-paths based random walks to generate node sequences to solve our problem.

Formally, a meta-path schema is in the form of $V_1 \xrightarrow{R_1} V_2 \xrightarrow{R_2} \cdots \xrightarrow{R_{q-1}} V_q$, wherein $R = R_1 \circ R_2 \cdots \circ R_q$ characterizes a new composite relation between its start type V_1 and the end type V_q [28]. Specifically, to characterize the relations among users, we define six types of meta-paths as shown in Table 1. These carefully designed meta-paths help us find a pair of entities that similar but could be distant from each other on the user-item bipartite network and the user-user social network. For instance, with the meta-path defined as P_5, given U_2 as the root (as shown in Fig. 1), after two steps, the walk reaches the items I_1 and I_3, which are not directly connected with U_2. In this way, we can characterize the similarity between distant user-item pairs.

The carefully designed meta-paths are used to conduct random walks to generate a number of node sequences. However, social relations are often noisy, thus we have to find out reliable sequences with biased probability. Here we show how meta-paths can be exploited to guide the random walks to generate biased node sequences - referred as the *social corpus*.

Given a meta-path schema $\mathcal{P} = V_1 \xrightarrow{R_1} V_2 \xrightarrow{R_2} \cdots \xrightarrow{R_{q-1}} V_q$, the transition probability at step k is defined as follows:

当前

下一步

$$p(v^{k+1}|v_n^k, \mathcal{P}) = \begin{cases} \frac{1}{|N_{k+1}(v_n^k)|} & (v^{k+1}, v_n^k) \in p \\ \frac{\psi(v^{k+1}, v_n^k)}{\sum_{v' \in N_{n+1}(v_n^k)} \psi(v', v_n^k)} & (v^{k+1}, v_n^k) \in t \\ 0 & (v^{k+1}, v_n^k) \notin E \end{cases} \quad (1)$$

where $v_n^k \in V_n$, $N_{n+1}(v_n^k)$ denotes the V_{n+1} type of neighborhood of node v_n^k, $\psi(v^{k+1}, v_n^k) = |N_{n+1}(v^{k+1}) \cap N_{n+1}(v_n^k)|$. It means, at each step of the random walk, the next node type is decided by the pre-defined meta-path $\mathcal{P}$. When $V_n = U$ and $V_{n+1} = I$ (or the inverse), we uniformly select the successor node. But in the case that $V_n = V_{n+1} = U$ (or the inverse), the successor node is chosen according to the number of overlapped neighbors with the current node. In other words, if the successor has a large number of shared friends with v_n^k, it is more likely to be selected, which is in light of the study [21] that the stronger the tie between two users are, the more their friends overlap. By doing so, we can obtain more reliable sequences and reduce the adverse consequences of noisy social relations. At last, it should be noted that the walk is recursive, that means, $V_{q+1} = V_2$.

Random walks based network representation learning [9] generally learns representations of nodes according to their co-occurrence in a context window. Therefore, it is necessary to distinguish positive and negative feedbacks. When conducting random walks, we only use the positive feedbacks and user relations to generate the positive social corpus. Likewise, the negative social corpus is based on negative feedbacks and user relations. These two social corpora, in which the complete preferences of users are encoded, will be used to generate different representations in the next step.

3.2 Learning Node Representations and Identifying Top-K Implicit Friends

The collected *social corpora* consist of different types of nodes and it is still not clear how to quantify the similarity among these nodes. Thus, we learn node embedding $Y \in \mathbb{R}^{|V| \times d}$ based on heterogeneous Skip-Gram [5], an extension of word2vec embeddings, to solve the problem. Formally, given a meta-path guided node sequence and the current node v^k, the objective function is:

$$\max_{\theta} \sum_{v \in V} \sum_{v_n^m \in C(v^k)} \log p(v_n^m | v^k; \theta), \quad (2)$$

where $C(v^k)$ is the context of v^k with the window size w; $p(v_n^m|v^k;\theta)$ is commonly defined as the heterogeneous softmax function:

$$p(v_n^m | v^k; \theta) = \frac{e^{y_{v_n^m} \cdot y_{v^k}}}{\sum_{v \in V_n} e^{y_v \cdot y_{v^k}}}. \quad (3)$$

Here y_v is the v^{th} row of Y, representing the embedding vector of node v, and V_n is the node set of type n in $\mathbf{H}$. The heterogeneous Skip-Gram maximizes the probability in terms of the local structures. But for each node in $C(v^k)$, it only considers nodes in the same type set instead of all the nodes, which makes it different from the conventional Skip-Gram model.

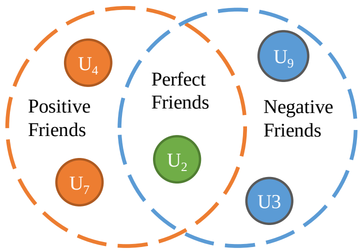

Figure 2: Different types of implicit friends.

The computation of $p(v_n^m|v^k;\theta)$ in Eq. (2) is time-consuming, which is difficult to be applied to large-scale networks. To accelerate the optimization, we adopt the negative sampling heuristic [20] for the learning task. Given the type of the node in $C(v^k)$ and the negative sample size I, we randomly select I nodes with the same type label from V for the construction of softmax and then update Eq. (2) by maximizing the following objective function:

$$O(Y) = \log\sigma(y_{v_n^m}\cdot y_{v^k}) + \sum_{i=1}^{I}\mathbb{E}_{v_n^i\sim P_n(v_n)}[\log\sigma(-y_{v_n^i}\cdot y_{v^k})], \quad (4)$$

where $\sigma(y) = \frac{1}{1+e^{-y}}$ and the sampling distribution $P_n(v_n)$ are determined by the node degree.

Note that we have two social corpora which depict the positive and negative user preferences, respectively. As a result, each user finally gets two representations. When the representation learning is done by performing stochastic gradient ascent on Eq. (4), we compute the cosine similarity for each pair of users w.r.t. their embeddings and identify the Top-K positive and negative implicit friends for each user. In particular, *Positive friends* are implicit friends who share similar positive preference with the current user while *negative friends* are the users who share similar negative preference. As the embedding vectors are continuous and dense, it enables us to compute the similarity for each pair of users, even if they are topologically distant from each other on social networks, which makes our model superior to other methods [7, 15, 29].

4 IF-BPR: BPR WITH IMPLICIT FRIENDS

In this section, we present our proposed IF-BPR model which incorporates the implicit friends into BPR and ranks social items based on different types of friends.

4.1 Model Assumption and Formulation

After the Top-K positive and negative implicit friends for each user are identified, we find that they are partially overlapped as expected (shown in Fig. 2). Then the overlapped part is named as *Perfect friends* since they share similar preference with the current user in both aspects. To this end, for each user, we have three types of implicit friends - implicit perfect friends, implicit positive friends, and implicit negative friends. Hence, the items that are not observed in the consumption history of the current user can be categorized in a fine-grained way. The proposed categorization is as follows.

- **Positive Items**: For all $u \in \mathcal{U}$, let P_u denote items consumed by u itself.
- **Joint Social Items**: Any item $i \in \mathcal{I}\backslash P_u \bigcup NS_u$ that has been consumed by at least one of u's perfect friends and received a positive feedback. This set is denoted as J_u.
- **Positive Social Items**: Any item $i \in \mathcal{I}\backslash P_u \bigcup J_u \bigcup NS_u$ that has been consumed by at least one of u's positive friends and received a positive feedback. We denote this set as PS_u.
- **Negative Social Items**: Any item $i \in \mathcal{I}\backslash P_u \bigcup J_u \bigcup PS_u$ that has been consumed by u itself or u's negative friends and received a negative feedback. This set is defined as NS_u.
- **Non-consumed Items**: The set N_u which contains the remaining items such that $\notin P_u \bigcup J_u \bigcup PS_u \bigcup NS_u$.

Obviously, $P_u \bigcup J_u \bigcup PS_u \bigcup NS_u \bigcup N_u = \mathcal{I}$ and they are disjoint with each other. Note that there is no Joint Negative Items which contains items that have been negatively rated by both of positive friends and negative friends or u itself since negative feedbacks are rather limited compared with positive feedbacks in real-world scenarios. Hence, elaborately categorizing negative feedbacks is prone to overfitting.

Generally, recommended items for users are presented as an ordered list. Therefore, items with higher rankings in the list are more likely to be noticed. BPR [24] is a typical one-class collaborative filtering algorithm which aims to model the preference-order for each user. However, BPR ignores relations among users in the social network. To fully take advantage of the social relations, a social BPR model (SBPR) [46] is built based on the assumption that users tend to assign higher ranks to items that their friends prefer. Specifically, SBPR extends BPR with the following relations:

$$x_{ui} \geq x_{uk}, x_{uk} \geq x_{uj}, i \in P_u, k \in SP_u, j \in N_u, \quad (5)$$

where $x_{u\cdot}$ denotes the preference score of user u on one of the candidate items, SP_u denotes the set of items the u did not express any positive feedback, but at least one of explicit friends did.

In our model, the above assumption is further expanded. Given u, P_u, J_u, PS_u, NS_u, and N_u, we aim to learn a ranking function for each user that can rank items by the following order:

$$\begin{aligned} f: & x_{ui} \geq x_{uj} \geq x_{uk} \geq x_{uc} \geq x_{un}, \\ & i \in P_u, j \in Ju, k \in PS_u, c \in N_u, n \in NS_u. \end{aligned} \quad (6)$$

This assumption can be easily interpreted in a way that perfect friends are supposed to have higher priority than positive friends while observed negative items should have a lower ranking than unobserved items.

Let $\Theta \equiv (\mathbf{Z}, \mathbf{Q})$ denote the latent user and item feature vectors, respectively. According to the model assumption, the optimization likelihood for each user can be represented as follows:

$$\begin{aligned} & \prod_{i\in P_u, j\in Ju}\mathcal{P}(x_{ui}\geq x_{uj}|\Theta) \prod_{j\in J_u, k\in PS_u}\mathcal{P}(x_{uj}\geq x_{uk}|\Theta) \\ & \prod_{k\in PS_u, c\in NC_u}\mathcal{P}(x_{uk}\geq x_{uc}|\Theta) \prod_{c\in NC_u, n\in N_u}\mathcal{P}(x_{uc}\geq x_{un}|\Theta), \end{aligned} \quad (7)$$

where $\mathcal{P}(x_{ui} \geq x_{uk}|\Theta)$ is defined as $\sigma(x_{ui} - x_{uk})$ and $x_{ui} = \mathbf{Z}_u^T\mathbf{Q}_i$.

4.2 Adaptive Refinement for Top-K Implicit Friends

In section 3, we have identified the Top-K (e.g. 100) positive and negative friends for each user. However, incorporating the same amount of implicit friends for each user may lead to a suboptimal solution because of the ubiquitous differences across different users. Warner et al. [33] revealed that positive social effect is not observed in all groups of users in the explicit social network. Users who have limited connection to other members are usually not influenced by their friends on the purchasing behaviors. In addition, the positive social effect is only moderately observed among connected users. In particular, to maintain the distinctiveness, highly connected users have a tendency to reduce their purchases of items when they observe that these items were consumed by their friends. Based on this study, we conjecture that different users are also influenced by their implicit friends in a rather diversified fashion. Consequently, assigning all users the same amount of implicit friends may not be a wise choice.

To tackle the above-mentioned problem, a feasible alternative is to adaptively refine the Top-K implicit friends for each user according to the performance of the model on the training data. As the Top-K implicit friends are ordered by the similarity with the current user, a similarity threshold t_u for the current user which varies with the proceeding of the training enables the adaptive refining. That means, if t_u becomes higher during the training, few implicit friends would be used to train the model. On the contrary, more implicit friends would be added. Given a threshold t_u, the importance of an implicit friend can be quantitatively measured using the following formula:

$$\mathrm{g} = \sigma(\frac{s_{uv} - t_u}{\bar{s}_u - t_u}), \tag{8}$$

where s_{uv} is the similarity between the current user u and one of its implicit friends v, and $\bar{s}_u$ is the average similarity of all positive implicit friends who have higher similarity than t_u with u.

To incorporate the threshold into the optimization such that our IF-BPR model is able to learn it adaptively, we add a coefficient $(1+\mathrm{g})$ into the probability that items in P_u are preferred over items in J_u. More specifically, we define:

$$\mathcal{P}(x_{ui} \geq x_{uj}|\Theta) = \sigma(\frac{x_{ui} - x_{uj}}{1 + \mathrm{g}}), i \in P_u, j \in J_u \tag{9}$$

where $(x_{ui} - x_{uj})$ is discounted by $(1 + \mathrm{g})$. The intuition is that the weight of item j is determined by both the similarity of the implicit friend v and the average similarity of all selected implicit friends of user u. Specifically, we use g to control the magnitude of perfect friends which are derived from both positive and negative friends.

4.3 Parameters Optimization

Generally, to avoid overfitting, zero-mean Gaussian priors are imposed on $\mathbf{Z}$ and $\mathbf{Q}$, generating a regularization term in the form of $\frac{\lambda_\Theta}{2}(||\mathbf{Z}||_F^2 + ||\mathbf{Q}||_F^2)$. Furthermore, we notice that, given the perfect friends $\mathcal{F}(u)$, user feature vectors follow a conditional distribution $\mathcal{N}(\mathbf{Z}_u|\hat{\mathbf{Z}}_u, \sigma_{\mathbf{Z}}^2\mathbf{I})$ with mean $\hat{\mathbf{Z}}_u$ and variance $\sigma_{\mathbf{Z}}^2$, and $\hat{\mathbf{Z}}_u$ is the average of feature vectors of users in $\mathcal{F}(u)$. The proposed social regularization is based on the assumption that a user's preference should be close to the average preferences of her friends, which is often not well exploited in the item ranking problem. Under this additional constraint, the influence of perfect friends is further expanded. In particular, if the adaptive refinement is not adopted because of the demand of fast running, imposing this constraint will significantly enable IF-BPR to avoid overfitting as a small amount of implicit friends only have rated a small proportion of items, which leads to small sets of J_u, PS_u and NS_u. With the rise of the amount of implicit friends, the influence of the social regularizer will be weaken. By taking negative log-form of the posterior probability, our model minimizes the following objective function, which is composed of order modeling and regularization terms:

$$\begin{aligned}\mathcal{L} = &-\sum_u \Bigg(\sum_{i \in P_u} \sum_{j \in J_u} \ln(\sigma(\frac{x_{ui} - x_{uj}}{1 + \mathrm{g}})) \\ &- \sum_{j \in J_u} \sum_{k \in PS_u} \ln(\sigma(x_{uj} - x_{uk})) - \sum_{k \in PS_u} \sum_{c \in N_u} \ln(\sigma(x_{uk} - x_{uc})) \\ &- \sum_{c \in N_u} \sum_{n \in NS_u} \ln(\sigma(x_{uc} - x_{un})) \Bigg) + \frac{\lambda_\Theta}{2}(||\mathbf{Z}||_F^2 + ||\mathbf{Q}||_F^2) \\ &+ \frac{\lambda_{\mathbf{Z}}}{2} \sum_u ||\mathbf{Z}_u - \frac{\sum_{u' \in \mathcal{F}(u)} \mathbf{Z}_{u'}}{|\mathcal{F}(u)|}||_F^2 + \frac{\lambda_t}{2} \sum_u ||t_u||^2.\end{aligned} \tag{10}$$

A local minimum of the log-likelihood function in Eq. (10) can be obtained by performing stochastic gradient descent on $\mathbf{Z}, \mathbf{Q}$ and t_u. Each time, we randomly sample a user and five items from the corresponding item sets to perform the optimization process.

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial \mathbf{Z}_u} &= -\frac{\frac{1}{1+\mathrm{g}} e^{-(\frac{x_{ui}-x_{uj}}{1+\mathrm{g}})}}{1 + e^{-(\frac{x_{ui}-x_{ujk}}{1+g})}}(\mathbf{Q}_i - \mathbf{Q}_j) - \frac{e^{-(x_{uj}-x_{uk})}}{1 + e^{-(x_{uj}-x_{uk})}}(\mathbf{Q}_j - \mathbf{Q}_k) \\ &\quad - \frac{e^{-(x_{uk}-x_{uc})}}{1 + e^{-(x_{uk}-x_{uc})}}(\mathbf{Q}_k - \mathbf{Q}_c) - \frac{e^{-(x_{uc}-x_{un})}}{1 + e^{-(x_{uc}-x_{un})}}(\mathbf{Q}_c - \mathbf{Q}_n) \\ &\quad + \lambda_Z(\mathbf{Z}_u - \frac{\sum_{u' \in \mathcal{F}(u)} \mathbf{Z}_{u'}}{|\mathcal{F}(u)|}) + \lambda_\Theta \mathbf{Z}_u, \\ \frac{\partial \mathcal{L}}{\partial \mathbf{Q}_i} &= -\frac{\frac{1}{1+\mathrm{g}} e^{-(\frac{x_{ui}-x_{uj}}{1+\mathrm{g}})}}{1 + e^{-(\frac{x_{ui}-x_{uj}}{1+g})}} \mathbf{Z}_u + \lambda_\Theta \mathbf{Q}_i, \\ \frac{\partial \mathcal{L}}{\partial \mathbf{Q}_j} &= \frac{\frac{1}{1+g} e^{-(\frac{x_{ui}-x_{uj}}{1+\mathrm{g}})}}{1 + e^{-(\frac{x_{ui}-x_{uj}}{1+\mathrm{g}})}} \mathbf{Z}_u - \frac{e^{-(x_{uj}-x_{uk})}}{1 + e^{-(x_{uj}-x_{uk})}} \mathbf{Z}_u + \lambda_\Theta \mathbf{Q}_j, \\ \frac{\partial \mathcal{L}}{\partial \mathbf{Q}_k} &= \frac{e^{-(x_{uj}-x_{uk})}}{1 + e^{-(x_{uj}-x_{uk})}} \mathbf{Z}_u - \frac{e^{-(x_{uj}-x_{uk})}}{1 + e^{-(x_{uj}-x_{uk})}} \mathbf{Z}_u + \lambda_\Theta \mathbf{Q}_k, \\ \frac{\partial \mathcal{L}}{\partial \mathbf{Q}_c} &= -\frac{e^{-(x_{uk}-x_{uc})}}{1 + e^{-(x_{uk}-x_{uc})}} \mathbf{Z}_u - \frac{e^{-(x_{uc}-x_{un})}}{1 + e^{-(x_{uc}-x_{un})}} \mathbf{Z}_u + \lambda_\Theta \mathbf{Q}_c, \\ \frac{\partial \mathcal{L}}{\partial \mathbf{Q}_n} &= -\frac{e^{-(x_{uc}-x_{un})}}{1 + e^{-(x_{uc}-x_{un})}} \mathbf{Z}_u + \lambda_\Theta \mathbf{Q}_n\end{aligned} \tag{11}$$

Meanwhile, to adaptively refine implicit friends for each user, an EM algorithm based update policy is adopted. Based on Eq. (10), we update t_u according to Eq. (12) (**E-step**). After each iteration on all users and sampled pairs, we filter out implicit friends which have a similarity below the threshold for each user and reconstruct the social item sets (**M-step**).

$$\frac{\partial \mathcal{L}}{\partial t_u} = \sum_{j \in J_u} -\Big(\frac{g(1-g)\frac{1}{1+g} e^{-(\frac{x_{ui}-x_{uj}}{1+g})}(x_{ui}-x_{uj})(s_{uv}-\bar{t}_u)}{1+e^{-(\frac{x_{ui}-x_{uj}}{1+s})}(\bar{t}_u-t_u)^2(1+g)^2|J_u|} - \frac{\lambda_t t_u}{|J_u|}\Big). \tag{12}$$

Compared with the original Social BPR [46] and other Social BPR based models [32, 43], IF-BPR enriches the social ranking assumption and proposes to refine the Top-K implicit friends for each user to ensure that the final selected implicit user relations could improve the social recommendation performance. Besides, existing Social BPR models sample negative pairs from unobserved data, which is ill-considered because unobserved item could also be the positive feedbacks. Instead, IF-BPR develops a wiser sampling strategy with the identified negative feedbacks and friends. The overview of the proposed method is illustrated in Fig. 2.

5 EXPERIMENTS AND RESULTS

In this section, we perform experiments to answer the following research questions: (1) Can adaptively learning the optimal amount of implicit friends improve recommendation performance? (2) Can IF-BPR show evident improvement when compared with other methods? (3) Can IF-BPR relieve the cold-start recommendation problem? (4) What are the roles implicit friends and explicit friends play for the social recommendation?

5.1 Experimental Designs

Datasets. Three common social recommendation datasets, LastFM [2], Douban [44], and Epinions [19] are used for experimental evaluations. It should be noted that as the main focus of this paper is to perform the top-N recommendation, for Epinions and Douban with a rating scale of 1 to 5, only the ratings of 4 and 5 are considered as the positive feedbacks and the ratings of 1 and 2 are considered as the negative feedbacks for the model training. For LastFM, the songs which were listened only for once by the current user are collected as the negative feedbacks. The detailed statistics of the used datasets are shown in Table 2. For all the datasets, we use 80% of the data as the training set, from which we randomly select 10% as the validation set. Specifically, the parameters of baseline methods are determined by their performance on the validation set. Then we conduct the experiments with 5-fold cross validation for 10 times and present the average performance.

Table 2: Dataset Statistics

Dataset	#Users	#Items	#Feedbacks	Density	#Relations
LastFM	1,892	17,632	92,834	0.278%	25,434
Douban	2,831	15,918	636,436	1.412%	35,624
Epinions	18,201	18,751	403,725	0.072%	203,720

Baseline Methods. To demonstrate the superiority of our approach, we compare IF-BPR with the most popular item ranking methods: Most-Pop (MP), BPR [24], SBPR [46], TBPR [32] and CUNE [43]. Besides, two network embedding methods, DeepWalk[23] and metapath2vec[5], are also trained to generate recommendations by computing the cosine similarities between users and items,

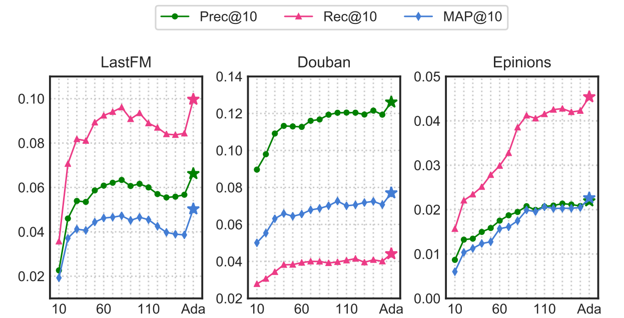

Figure 3: Comparison of adaptive learning and Top-K. (Star denotes the result of IF-BPR$^+$).

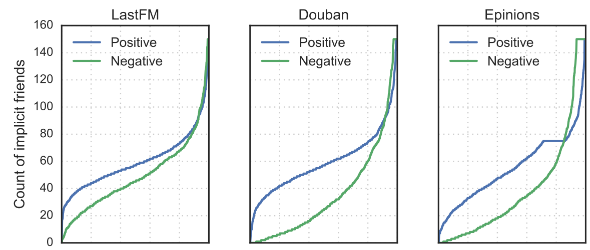

Figure 4: The number of implicit friends of users.

and all the node sequences used in metapath2vec are the same as those used in IF-BPR. Among these methods, SBPR directly uses the explicit relations, TBPR divides explicit relations into strong and weak sets, and CUNE is the most similar approach to our proposed method as it also leverages network embedding techniques to obtain the Top-K implicit friends. However, CUNE identifies the Top-K implicit friends from a homogeneous network (i.e., the collaborative user network derived from the user-item bipartite network), while our developed method models the whole system as a heterogeneous information network, and the comparison with CUNE can further reveal the advantages of the proposed IF-BPR model.

Evaluation Metrics. Two relevance-based metrics - *Precision@K* and *Recall@K*, and one ranking-based metric - *MAP@K* (Mean average precision) are used to measure the recommendation performance of our proposed model and various baseline methods.

Configuration. For all baseline models, the regularization coefficient λ_Θ is set as 0.01 and the dimension of latent features d is specified as 20. For the two network embedding based model, CUNE and IF-BPR, the number of walks is $n = 20$, the length of each walk is $l = 20$, the dimension of embedding is $Y = 25$, the window size is $w = 5$, the number of negative samples is $M = 5$, and the amount of implicit friends for CUNE is decided by its optimal performance on three datasets. For IF-BPR, the social regularization coefficient λ_s is empirically specified as 0.2 and the regularization coefficient for similarity threshold is $\lambda_t = 0.01$. Moreover, In our experiments, 40% of node sequences in IF-BPR are generated by the meta-path $U \xrightarrow{P} I \xleftarrow{P} U$, with the remaining part shared by other paths.

Table 3: Performance comparison of our methods and other methods.

Dataset	Metric	MP	BPR	SBPR	TBPR	CUNE	DeepWalk	MP2vec	IF-BPR	**IF-BPR$^+$**	Improv.
LastFM	Prec@10	3.090%	4.910%	4.846%	5.321%	4.770%	1.906%	2.138%	6.338%	**6.622**%	24.450%
	Prec@20	2.037%	2.975%	3.037%	3.224%	3.080%	1.185%	1.345%	3.783%	**3.931**%	21.923%
	Rec@10	4.674%	7.471%	7.184%	8.038%	7.144%	2.706%	3.042%	9.609%	**9.972**%	24.060%
	Rec@20	6.130%	9.023%	8.991%	9.437%	9.192%	3.371%	3.824%	11.363%	**11.804**%	25.082%
	MAP@10	0.02264	0.03367	0.03213	0.04194	0.03681	0.01245	0.01383	0.04723	**0.05035**	20.052%
	MAP@20	0.02399	0.03522	0.03406	0.04268	0.03909	0.01306	0.01457	0.04945	**0.05262**	23.289%
Douban	Prec@10	10.256%	10.366%	8.806%	9.578%	11.442%	0.704%	1.188%	12.156%	**12.612**%	10.575%
	Prec@20	7.426%	8.103%	6.909%	7.188%	8.377%	0.556%	0.997%	8.879%	**9.085**%	8.452%
	Rec@10	3.273%	3.551%	2.705%	3.347%	3.874%	0.109%	0.158%	4.086%	**4.417**%	14.017%
	Rec@20	4.752%	5.295%	4.228%	4.863%	5.498%	0.173%	0.284%	5.816%	**6.063**%	10.276%
	MAP@10	0.06162	0.05636	0.04517	0.05115	0.06455	0.00259	0.00524	0.07241	**0.07702**	19.318%
	MAP@20	0.04025	0.03921	0.03062	0.03413	0.04263	0.00172	0.00342	0.04867	**0.05133**	20.408%
Epinions	Prec@10	1.174%	1.732%	1.828%	1.960%	2.040%	1.378%	1.127%	2.136%	**2.195**%	7.598%
	Prec@20	0.852%	1.382%	1.281%	1.379%	1.417%	1.007%	0.904%	1.441%	**1.482**%	4.587%
	Rec@10	2.589%	4.029%	3.981%	3.997%	3.923%	2.929%	2.675%	4.277%	**4.541**%	13.610%
	Rec@20	3.541%	5.425%	5.521%	5.419%	5.301%	3.990%	3.610%	5.678%	**5.957**%	7.897%
	MAP@10	0.01312	0.01877	0.01820	0.01942	0.01937	0.01366	0.01160	0.02031	**0.02264**	16.580%
	MAP@20	0.01322	0.01883	0.01837	0.01953	0.01944	0.01382	0.01161	0.02056	**0.02244**	14.900%

5.2 Adaptive Refining *vs.* Top-K

Intuitively, the assumption that all users share the same amount of Top-K implicit friends may lead to suboptimal recommendation performance. That is the reason why we propose to adaptively refine the Top-K implicit friends for each user. In this part, to ascertain that adaptive refining is necessary, we compare the performance of two variants of IF-BPR. The one with the refinement is named IF-BPR$^+$ while the other without this procedure is still named IF-BPR. For IF-BPR, We change the number of implicit friends (Top-K) in the range of [10, 150] with a step of 10. For IF-BPR$^+$, the initial similarity threshold of each user is the median similarity of Top-150 implicit friends.

As shown in Fig. 3, on Douban and Epinions, the performance of IF-BPR increases with the rise of the number of implicit friends, and then it gradually reaches a stable state. On LastFM, IF-BPR initially gets a poor performance and then the performance monotonously increases to the best when $k = 80$. Afterwards, it starts to drop. The reason why IF-BPR gets unsatisfied performance when K is small is due to the overfitting since the user-item pairs in J_u, PF_u, and NF_u are limited. Obviously, IF-BPR$^+$ outperforms IF-BPR by a fair margin, which shows the effectiveness of adaptively refining the Top-K implicit friends for each user.

Furthermore, to confirm that IF-BPR$^+$ learns a different number of implicit friends for each user, we randomly extract 1,000 users from each dataset and draw the number of implicit friends when IF-BPR$^+$ finishes the running. From Fig. 4 we can see that different users do have a different number of implicit friends.

5.3 Recommendation Performance

The recommendation performance of different methods is shown in Table 3. We can make the following observations from the table:

(1) In all cases, our proposed models IF-BPR and IF-BPR$^+$ outperform all the compared baseline methods. Specifically, on two denser datasets LastFM and Douban, IF-BPR and IF-BPR$^+$ achieve much better recommendation performance than other methods w.r.t. *Precision@k*, *Recall@k* and *MAP@k*. The relative improvements (calculated by comparing with the second best performance) vary in the range of 4.587% to 25.082%.

(2) Two implicit friends based recommendation approaches, CUNE and IF-BRP$^{(+)}$ achieve better recommendation performance than explicit friends based recommendation methods in most cases. The result can be explained that the embedding learning plays an important role in quantifying the similarity of topologically distant users for the recommendation and implicit friends can closely reflect the current user's preference compared with explicit friends. By contrast, two network embedding methods show a poor performance, especially on the dataset of Douban. We believe this is because these methods do not model the preference order. That is why we propose IF-BPR to enhance recommendation instead of using embeddings to generate personalized recommendations directly.

(3) In comparison to BPR, SBPR does not show any evident improvements in half of the cases. In particular, on Douban dataset which has a higher feedback density, BPR even performs better than SBPR and TBPR. The potential reason is that users' own feedbacks are enough to fit their preferences. As a consequence, noises in social networks may lower the performance. It once again confirms the fact that explicit social relations are not always helpful for the recommendation.

5.4 Recommendation for Cold-Start Users

The real challenge for social recommendation is to perform the accurate recommendation for the cold-start users who have a limited amount of feedbacks or ratings in the system. In this part, we conduct experiments on users who have feedbacks less than 10 to validate if IF-BPR can alleviate the cold-start problem.

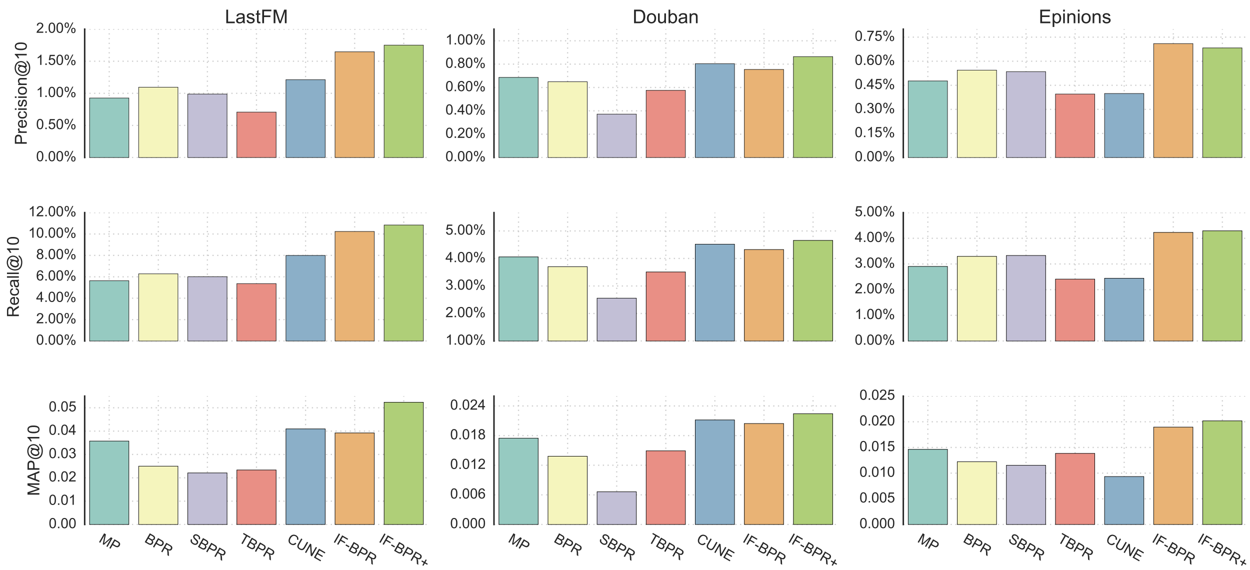

Figure 5: Evaluation on cold-start users with less than 10 feedbacks or ratings.

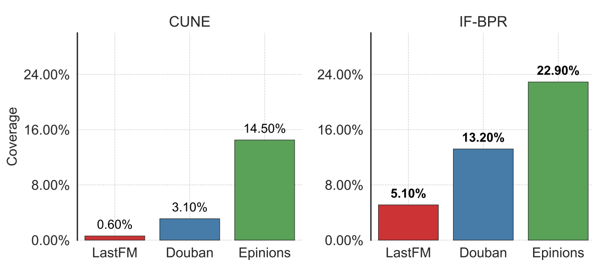

Figure 6: Coverage for cold-start users.

From Fig. 5, we can see that IF-BPR$^+$ leads to more accurate recommendations than other baseline methods in almost all cases. To our surprise, SBPR and TBPR are inferior to BPR on Douban. The reason may be that social relations are often too noisy to be directly used to improve the recommendation performance and cold-start users are also cold with regard to the social relations. As for CUNE and IF-BPR, both of them make use of enough implicit friends for the recommendation, which alleviates the cold-start problem with a significant improvement in recommendation performance.

However, why IF-BPR$^+$ outperforms CUNE is still not explored yet. We then turn our attention to investigate the network structure. In addition to the adaptive learning procedure, it is easy to understand that the centrality of cold-start users are often very low as few other users connect to them. In IF-BPR$^+$, we design meta-paths P_2 to P_6 to alleviate this issue by allowing the nodes with low centrality reach other nodes other than the directed neighbors in $\mathbf{G_r}$. But in CUNE, the random walk only traverses the collaborative user network derived from $\mathbf{G_r}$. Hence, the nodes with lower centrality scores are less likely to appear in the social corpora. To further corroborate these assumptions, we randomly select 1,000 nodes from the social corpora generated by CUNE and IF-BPR$^+$, respectively. As is shown in Fig. 6, IF-BPR$^+$ covers more cold-start users on all these three datasets. Consequently, the cold-start users are more likely to obtain a better embedding representation as they appear more in the social corpora. The above explorations explain why IF-BPR$^+$ is superior to CUNE in practice.

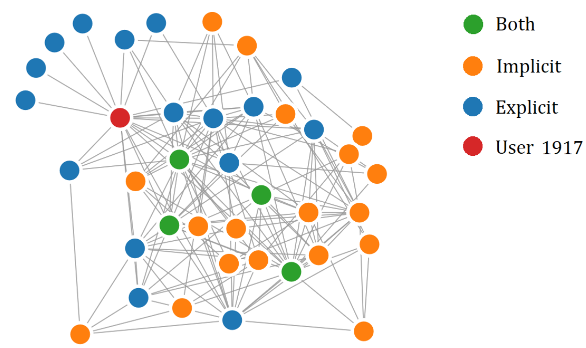

Figure 7: The ego social network of the user with ID '1917'.

5.5 Explicit Friends *vs* Implicit Friends

Above experiments validate the effectiveness of leveraging the implicit friends for the recommendation. It is natural for us to ask what are the roles the explicit friends and implicit friends play in the social recommendation? First, we examine whether the implicit friends and explicit friends overlap with each other. As expected, we only witness a less than 25% overlap. To illustrate the result, we randomly extract a user with id '1917' and her explicit and perfect implicit friends from Epinions, and then visualize the topology of the ego social network of user '1917'. According to Fig. 7, we can notice that overlapped explicit friends are highly connected with other explicit friends, whereas the nodes that are almost isolated with other explicit friends are less likely to be the implicit friends. If the isolated explicit friends are labeled as unreliable in terms of user interactions, this phenomenon can be a strong evidence showing that implicit friends are less noisy compared with the explicit friends. Considering that implicit friends are those who have similar tastes, we can draw a conclusion that the implicit friends could be complementary for the reliable explicit friends for the social recommendation.

Generally, social relations approximately follow the power-law distribution. That is to say, a small fraction of users generate most of the observed links. As social recommender systems directly make use of the observed links, the power-law distribution implies that conventional social recommender systems are actually measuring the similarity between users with high degree centrality and cannot well handle the tail users during the recommendation. As a consequence, it lowers the diversity of recommendations. In Fig. 8, we draw the follower distributions of explicit friends and implicit friends. As can be seen, the links of implicit friends are more evenly distributed over the whole crowds. This is another reason that can explain why IF-BPR is more effective than conventional social recommender systems.

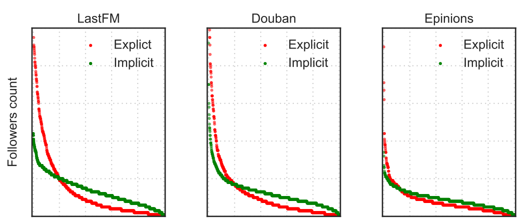

Figure 8: Follower relationship distribution (1000 users are randomly selected from each dataset).

6 RELATED WORK

In the early stage, studies on social recommendation mainly focused on discussing how to fully take advantage of the explicit social relations to improve the recommendation performance. Among those, Ma et al. [17] proposed a model to co-factorize the rating matrix and the relation matrix by connecting social information and rating information through the shared user latent feature. An ensemble based method [16] is proposed which assumes that the preferences of users are largely determined by the tastes of their friends. Later on, matrix factorization based social recommender systems are also developed [11, 18] with the same principle. Based on these works, Liu et al. [14] later introduced a model based on Bayesian inference to enable the recommendation over online social networks. Unlike other methods, in the work [10], two trust models were proposed. In contrast to other social recommenders, this work considers that both the trustors and trustees can affect usersâĂŹ preferences. In addition, various one-class collaborative filtering models [22] for social recommendation are also well studied. For example, in [46], the authors proposed to adapt BPR [24] to social personalized ranking with the observation that the preference lists of friends are similar to each other. In [3], a probabilistic model is developed to incorporate social relations into traditional factorization methods.

Follow-up findings showed that the direct usage of explicit social relations may result in an inferior performance [30]. Subsequent studies then resorted to identifying credible relations from online social networks. [36] proposed to find a wise group of experts in social networks to solve specific tasks. In the research [6], the authors argued that people trusting each other may not always share similar preferences. In their work, the original single-aspect trust information is decomposed into four general trust aspects. Shortly after, Wang et al. [32] proposed to leverage strong and weak social ties among two nodes for the social recommendation. In the work [31], the above- mentioned method is extended by learning personalized similarity thresholds for different users in order to differentiate close friends from casual acquaintances. Besides, there are a few studies proposed trust metrics to search for reliable implicit friends by computing and predicting trust scores between users based on their interactions [1, 7, 29]. Furthermore, inspired by the recent advances in network embedding learning such as DeepWalk [23], an embedding based social recommender system called CUNE is developed by identifying credible semantic friends on the constructed collaborative user network [43]. Despite the fact that aforementioned models have achieved decent improvements, few of them model the whole system as a heterogeneous network for the social recommendation, through which we can capture the similarity of users that are implicit from each other on the social network. It also should be mentioned that our work is similar but distinct from CUNE as we are the first to study the social recommendation by learning embedding representations of nodes on a heterogeneous network while CUNE focuses on homogeneous networks.

7 CONCLUSION

This paper aims to identify implicit friends toward the discovery of more credible user relations for the social recommendation. Inspired by the recent advances of network embedding, we propose a novel social recommendation method called IF-BPR. Our model first generates a sequence of nodes under the guidance of specifically designed meta-paths in which both the user-item and user-user relations are maximally encoded, and then map each user into a latent feature space by embedding representation learning. In this way, it enables us to find the Top-K implicit friends that are not necessarily connected with each other on the social network, and the obtained Top-K implicit friends are further exploited for the recommendation via an adaptive social Bayesian personalized ranking approach. Extensive experiments show that IF-BPR significantly improves the quality of social recommendation, even for the cold-start users. Moreover, the analysis reveals the reason why IF-BPR shows a remarkable improvement and proves that the way we search for implicit friends is reasonable and promising.

ACKNOWLEDGEMENTS.

This research is supported by National Science Foundation (NSF) under grant number 1614576, the Graduate Scientific Research and Innovation Foundation of Chongqing (cys17035) and China Scholarship Council.

REFERENCES

[1] Mohammad Mahdi Azadjalal, Parham Moradi, Alireza Abdollahpouri, and Mahdi Jalili. 2017. A trust-aware recommendation method based on Pareto dominance and confidence concepts. *Knowledge-Based Systems* 116, C (2017), 130–143.

[2] Iván Cantador, Peter Brusilovsky, and Tsvi Kuflik. 2011. 2nd Workshop on Information Heterogeneity and Fusion in Recommender Systems (HetRec 2011). In *Proceedings of the 5th ACM conference on Recommender systems (RecSys 2011)*. ACM, New York, NY, USA.

[3] Allison JB Chaney, David M Blei, and Tina Eliassi-Rad. 2015. A probabilistic model for using social networks in personalized item recommendation. In *Proceedings of the 9th ACM Conference on Recommender Systems*. ACM, 43–50.

[4] Kewei Cheng, Jundong Li, and Huan Liu. 2017. Unsupervised feature selection in signed social networks. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 777–786.

[5] Yuxiao Dong, Nitesh V Chawla, and Ananthram Swami. 2017. metapath2vec: Scalable representation learning for heterogeneous networks. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 135–144.

[6] Hui Fang, Yang Bao, and Jie Zhang. 2014. Leveraging decomposed trust in probabilistic matrix factorization for effective recommendation. In *Twenty-Eighth AAAI Conference on Artificial Intelligence*. 30–36.

[7] Soude Fazeli, Babak Loni, Alejandro Bellogin, Hendrik Drachsler, and Peter Sloep. 2014. Implicit vs. explicit trust in social matrix factorization. In *Proceedings of the 8th ACM Conference on Recommender systems*. ACM, 317–320.

[8] Evgeny Frolov and Ivan V Oseledets. 2016. Fifty Shades of Ratings: How to Benefit from a Negative Feedback in Top-N Recommendations Tasks. *Proceedings of the 10th ACM Conference on Recommender systems* (2016), 91–98.

[9] Palash Goyal and Emilio Ferrara. 2018. Graph embedding techniques, applications, and performance: A survey. *Knowledge Based Systems* (2018).

[10] Guibing Guo, Jie Zhang, and Neil Yorke-Smith. 2015. TrustSVD: collaborative filtering with both the explicit and implicit influence of user trust and of item ratings. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*. 123–129.

[11] Mohsen Jamali and Martin Ester. 2010. A matrix factorization technique with trust propagation for recommendation in social networks. In *ACM Conference on Recommender Systems*. 135–142.

[12] Yehuda Koren, Robert M Bell, and Chris Volinsky. 2009. Matrix Factorization Techniques for Recommender Systems. *IEEE Computer* 42, 8 (2009), 30–37.

[13] Jundong Li, Harsh Dani, Xia Hu, Jiliang Tang, Yi Chang, and Huan Liu. 2017. Attributed network embedding for learning in a dynamic environment. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. ACM, 387–396.

[14] Juntao Liu, Caihua Wu, and Wenyu Liu. 2013. Bayesian Probabilistic Matrix Factorization with Social Relations and Item Contents for recommendation. *Decision Support Systems* 55, 3 (2013), 838–850.

[15] Hao Ma. 2013. An experimental study on implicit social recommendation. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*. 73–82.

[16] Hao Ma, Irwin King, and Michael R Lyu. 2009. Learning to recommend with social trust ensemble. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*. 203–210.

[17] Hao Ma, Haixuan Yang, Michael R Lyu, and Irwin King. 2008. Sorec: social recommendation using probabilistic matrix factorization. In *Proceedings of the 17th ACM conference on Information and knowledge management*. ACM, 931–940.

[18] Hao Ma, Dengyong Zhou, Chao Liu, Michael R Lyu, and Irwin King. 2011. Recommender systems with social regularization. In *Proceedings of the fourth ACM international conference on Web search and data mining*. ACM, 287–296.

[19] Paolo Massa and Paolo Avesani. 2007. Trust-aware recommender systems. In *Proceedings of the 2007 ACM conference on Recommender systems*. ACM, 17–24.

[20] Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S Corrado, and Jeffrey Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. *neural information processing systems* (2013), 3111–3119.

[21] J. P. Onnela, J SaramÃďki, J HyvÃűnen, G SzabÃş, D Lazer, K Kaski, J KertÃĺsz, and A. L. BarabÃąsi. 2006. Structure and tie strengths in mobile communication networks. *Proceedings of the National Academy of Sciences of the United States of America* 104, 18 (2006), 7332.

[22] Rong Pan, Yunhong Zhou, Bin Cao, Nathan N. Liu, Rajan Lukose, Martin Scholz, and Qiang Yang. 2008. One-Class Collaborative Filtering. In *Eighth IEEE International Conference on Data Mining*. 502–511.

[23] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 701–710.

[24] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian personalized ranking from implicit feedback. In *Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence*. AUAI Press, 452–461.

[25] Paul Rozin and Edward B Royzman. 2001. Negativity Bias, Negativity Dominance, and Contagion. *Personality and Social Psychology Review* 5, 4 (2001), 296–320.

[26] Rashmi R Sinha, Kirsten Swearingen, et al. 2001. Comparing recommendations made by online systems and friends.. In *DELOS workshop: personalisation and recommender systems in digital libraries*, Vol. 106.

[27] Yizhou Sun and Jiawei Han. 2012. Mining heterogeneous information networks: principles and methodologies. *Synthesis Lectures on Data Mining and Knowledge Discovery* 3, 2 (2012), 1–159.

[28] Yizhou Sun, Jiawei Han, Xifeng Yan, Philip S Yu, and Tianyi Wu. 2011. Pathsim: Meta path-based top-k similarity search in heterogeneous information networks. *Proceedings of the VLDB Endowment* 4, 11 (2011), 992–1003.

[29] Seyed Mohammad Taheri, Hamidreza Mahyar, Mohammad Firouzi, Elahe Ghalebi K, Radu Grosu, and Ali Movaghar. 2017. Extracting implicit social relation for social recommendation techniques in user rating prediction. In *Proceedings of the 26th International Conference on World Wide Web Companion*. International World Wide Web Conferences Steering Committee, 1343–1351.

[30] Jiliang Tang, Xia Hu, and Huan Liu. 2013. Social recommendation: a review. *Social Network Analysis and Mining* 3, 4 (2013), 1113–1133.

[31] Xin Wang, Steven C. H Hoi, Martin Ester, Jiajun Bu, and Chun Chen. 2017. Learning Personalized Preference of Strong and Weak Ties for Social Recommendation. In *The International Conference*. 1601–1610.

[32] Xin Wang, Wei Lu, Martin Ester, Can Wang, and Chun Chen. 2016. Social Recommendation with Strong and Weak Ties. In *ACM International on Conference on Information and Knowledge Management*. 5–14.

[33] Steven Philip Warner. 2013. Do friends influence purchases in a social network? *Business & Economy* (2013).

[34] Nhat Lam Xuan, Thuc Vu, Trong Duc Le, and Anh Duc Duong. 2008. Addressing cold-start problem in recommendation systems. In *International Conference on Ubiquitous Information Management and Communication*. 208–211.

[35] Hongzhi Yin, Hongxu Chen, Xiaoshuai Sun, Hao Wang, Yang Wang, and Quoc Viet Hung Nguyen. 2017. SPTF: a scalable probabilistic tensor factorization model for semantic-aware behavior prediction. In *Data Mining (ICDM), 2017 IEEE International Conference on*. IEEE, 585–594.

[36] Hongzhi Yin, Bin Cui, and Yuxin Huang. 2011. Finding a wise group of experts in social networks. In *International Conference on Advanced Data Mining and Applications*. Springer, 381–394.

[37] Hongzhi Yin, Zhiting Hu, Xiaofang Zhou, Hao Wang, Kai Zheng, Quoc Viet Hung Nguyen, and Shazia Sadiq. 2016. Discovering interpretable geo-social communities for user behavior prediction. In *Data Engineering (ICDE), 2016 IEEE 32nd International Conference on*. IEEE, 942–953.

[38] Hongzhi Yin, Weiqing Wang, Hao Wang, Ling Chen, and Xiaofang Zhou. 2017. Spatial-aware hierarchical collaborative deep learning for POI recommendation. *IEEE Transactions on Knowledge and Data Engineering* 29, 11 (2017), 2537–2551.

[39] Hongzhi Yin, Xiaofang Zhou, Bin Cui, Hao Wang, Kai Zheng, and Quoc Viet Hung Nguyen. 2016. Adapting to User Interest Drift for POI Recommendation. *IEEE Transactions on Knowledge and Data Engineering* 28, 10 (2016), 2566–2581.

[40] Junliang Yu, Min Gao, Wenge Rong, Yuqi Song, and Qingyu Xiong. 2017. A Social Recommender Based on Factorization and Distance Metric Learning. *IEEE Access* 5, 99 (2017), 21557–21566.

[41] Junliang Yu, Min Gao, Yuqi Song, Zehua Zhao, Wenge Rong, and Qingyu Xiong. 2017. Connecting Factorization and Distance Metric Learning for Social Recommendations. In *International Conference on Knowledge Science, Engineering and Management*. Springer, 389–396.

[42] Xiao Yu, Xiang Ren, Yizhou Sun, Bradley Sturt, Urvashi Khandelwal, Quanquan Gu, Brandon Norick, and Jiawei Han. 2013. Recommendation in heterogeneous information networks with implicit user feedback. In *ACM Conference on Recommender Systems*. 347–350.

[43] Chuxu Zhang, Lu Yu, Yan Wang, Chirag Shah, and Xiangliang Zhang. 2017. Collaborative User Network Embedding for Social Recommender Systems. In *Proceedings of the 2017 SIAM International Conference on Data Mining*. SIAM, 381–389.

[44] Guoshuai Zhao, Xueming Qian, and Xing Xie. 2016. User-Service Rating Prediction by Exploring Social Users' Rating Behaviors. *IEEE Transactions on Multimedia* 18, 3 (2016), 496–506.

[45] Huan Zhao, Quanming Yao, Jianda Li, Yangqiu Song, and Dik Lun Lee. 2017. Meta-Graph Based Recommendation Fusion over Heterogeneous Information Networks. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 635–644.

[46] Tong Zhao, Julian McAuley, and Irwin King. 2014. Leveraging social connections to improve personalized ranking for collaborative filtering. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*. ACM, 261–270.