PAPER

Recommender System Using Implicit Social Information

Yusheng LI[†a)], ***Member***, **Meina SONG**[†], ***and*** **Haihong E**[†], ***Nonmembers***

SUMMARY Social recommendation systems that make use of the user's social information have recently attracted considerable attention. These recommendation approaches partly solve cold-start and data sparsity problems and significantly improve the performance of recommendation systems. The essence of social recommendation methods is to utilize the user's explicit social connections to improve recommendation results. However, this information is not always available in real-world recommender systems. In this paper, a solution to this problem of explicit social information unavailability is proposed. The existing user-item rating matrix is used to compute implicit social information, and then an ISRec (implicit social recommendation algorithm) which integrates this implicit social information and the user-item rating matrix for social recommendation is introduced. Experimental results show that our method performs much better than state-of-the-art approaches; moreover, complexity analysis indicates that our approach can be applied to very large datasets because it scales linearly with respect to the number of observations in the matrices.

key words: *recommender systems, implicit social relationships, item relations, probabilistic matrix factorization*

1. Introduction

Due to the exponential growth of information on the Web, there is a need for tools to help users select desired online information. A recommender system is an information filtering technique that suggests information items (movies, books, music, news, Web pages, images, etc.) that are likely to interest the user. Collaborative Filtering (CF) is one of the most widely used techniques for recommender systems. It leverages the user-item preference (or rating) patterns derived from a large amount of historic data to make the recommendation. However, recommender systems based on collaborative filtering suffer from the following inherent weaknesses: (1) Recommendation performances deteriorate when available ratings are very sparse, but data sparsity is a common phenomenon in recommender systems [19], and the density of available ratings in commercial recommender systems is often less than 1%. (2) Almost all traditional recommendation algorithms employ only user-item rating matrix information and assume that users are independent and identically distributed (i.i.d). This assumption ignores social interactions or connections between users but, in reality, we often turn to friends we trust for movie, music, or book recommendations, and our tastes and preferences are easily affected by our friends.

In order to overcome the problems mentioned above, there has been much recent interest in social recommendation studies [9]–[12] which aim to leverage systematically the social relationships among users, as well as their past behavior, for automatic recommendations. Social recommendation has already shown that it can improve the performance of recommendation systems; however, there remains the problem that explicit social connection information is not always available in many real-world recommender systems. Only a few Web sites have implemented social or trust mechanisms (such as Epinions and Douban). This problem greatly limits the implementation of social recommendation methods.

As social recommendations improve recommendation performance, it would be advantageous to use this method in more real world recommender systems, even when the user's social connection information is unavailable. To achieve this goal, this paper proposes an alternative method. When the user's explicit social information is unavailable, we use the existing user-item rating matrix to compute the similarity between users. Following the intuition that similar users often have the same or similar tastes, we can take this similarity as implicit social connection information between users, forming an implicit social network. In this paper, the relationship between items has also been considered because item relations can be important factors in many recommendation scenarios. In order to use the implicit social information, we propose a method based on probabilistic factor analysis which integrates the implicit social network structure and the user-item rating matrix. In our method, we connect the user's implicit social matrix, the item relations matrix, and the user-item rating matrix through shared user latent feature space and item latent feature space. By performing factor analysis based on probabilistic matrix factorization, the low-rank user latent feature space and item latent feature space are learned in order to make implicit social recommendations.

Experiment results on the MovieLens dataset show that our method outperforms state-of-the-art collaborative filtering algorithms in cases where explicit social information is unavailable and, on the Epinions dataset, show that our method, using implicit social information, performs only slightly worse than using explicit social information in cases where this information is available. Moreover, complexity analysis indicates that our approach can be applied to very

Manuscript received May 29, 2014.
Manuscript revised September 19, 2014.
Manuscript publicized October 29, 2014.

[†]The authors are with the PCN&CAD Center Lab, Beijing University of Posts and Telecommunications, Beijing 100876, China.
a) E-mail: liyusheng0100@163.com

DOI: 10.1587/transinf.2014EDP7174

large datasets, since it scales linearly with the number of observations.

The remainder of this paper is organized as follows. In Sect. 2, we provide an overview of several major approaches for recommender systems and some related work. Section 3 presents our implicit social recommender system. The experimental results and analysis are presented in Sect. 4, followed by the conclusions and future work in Sect. 5.

2. Related Work

In this section, we review several major approaches for recommender systems, including (1) traditional recommender systems that are mainly based on collaborative filtering techniques and (2) social recommender systems that have recently drawn attention.

2.1 Collaborative Recommendation

The most popular algorithm in recommendation system is collaborative filtering. Generally, collaborative filtering systems are divided into two categories, i.e., memory-based [2]–[5] and model-based [1], [6]–[8]. Memory-based methods are further divided by their focus on finding similar users [2], [4], or items [3], [5], for recommendations. User-based approaches predict active user ratings based on those of similar users, and item-based approaches predict active user ratings based on computed information of items similar to those chosen by the active user. User-based and item-based approaches often use the PCC algorithm [13] and the VSS (vector space similarity) algorithm [14] as similarity computation methods.

Model-based filtering systems assume that users build up clusters, based on similar behavior, in rating items. A model is learned based on patterns recognized in user rating behaviors using clustering, Bayesian networks, and other machine learning techniques [20], [21]. Recently, several matrix factorization methods [1], [17], [18], [22] have been proposed for collaborative filtering. These methods focus on fitting the user-item rating matrix using low-rank approximations, and use it to make further predictions. Matrix factorization methods and low-dimensional factor models are very efficient in training, since they assume that only a small number of factors influence preferences in the user-item rating matrix, and that a user's preference vector is determined by the way each factor applies to that user.

There are also hybrid approaches which combine collaborative with content-based methods, or with different variants of other collaborative methods.

2.2 Social Recommender Systems

Traditional recommender systems have been extensively studied and developed, both in academia and in industry, but they are all based on the assumption that users are independent and identically distributed, and all ignore the relationships between users. Researchers have recently started to analyze social recommender systems, taking into account the social networks between users to improve recommendation quality. The belief here is that users linked with each other in social networks tend to share certain common interests or to have similar tastes, which can help increase recommendation accuracy.

There have been some recent forays into social recommendations [9], [10], [23]–[25], all based on the assumption that any pair of friends in a social network will have similar interests. The studies [24], [25] incorporate this network-based similarity property between users into a state-of-the-art matrix factorization recommendation approach. The authors of one study [9] have proposed a social recommendation framework, which is a probabilistic matrix factorization-based method which learns the user and item latent feature spaces by employing a user social network and a user-item rating matrix, simultaneously and seamlessly. In another study [10], two social regularization methods have been proposed, constraining the matrix factorization objective function with user social regularization terms. Those approaches have shown reasonable improvement over state-of-the-art recommendation approaches which do not utilize social factors.

The explicit user information used in social recommendation is not always available; this problem greatly limits the utilization of social recommendation, but there are alternative approaches that use other information to improve recommendation performance. Tag information can be used to compute the neighbors of users and items, then the neighbors' information is used to improve the latent factor of users and items [15]. Degrees of similarity can be used to divide users and items into two categories: similar and dissimilar users and items, to extend information to the recommender algorithm [16]. In this paper, a social recommendation algorithm is proposed which differs from previous work in three aspects: (1) Our method incorporates item relation information, based on traditional social recommender algorithms; (2) Our method is interpreted using a probabilistic factor analysis model by utilizing a user-item rating matrix and implicit user and item social information matrix together, thus combining the best elements of both types of information; (3) Our framework makes it easy to use implicit information, so we use the entire implicit social information between users and items. We believe that the computed implicit information is usefully for recommendation results. The experiment results confirm our assumptions.

3. Implicit Social Recommender Algorithm

In this section, we first introduce a probabilistic matrix factorization-based latent factor analysis method. Then, we illustrate how to construct implicit social information and leverage this in cases where explicit user social information is not available.

3.1 User-Item Rating Matrix Factorization

Considering the user-item rating matrix, suppose we have m users and n items. The user-item rating matrix is denoted as R, and the element r_{ij} in R means the rating to item i_j by user u_i, where values of r_{ij} are within the range [0,1]. Actually, most recommender systems use integer rating values from R_{min} to R_{max} to represent the user's judgement in items; we use the function $f(x) = (x - R_{min})/(R_{max} - R_{min})$ as the mapping function to map the original rating values to values in the interval [0,1]. We use $U \in R^{l\times m}$ and $V \in R^{l\times n}$ to denote the user latent feature matrix and the item latent feature matrix, respectively, with column vectors U_i and V_j denoting user-specific and item-specific latent feature vectors, respectively. We define the conditional distribution over the observed ratings as:

$$P(R|U, V, \sigma_R^2) = \prod_{i=1}^{m}\prod_{j=0}^{n}[N(r_{ij}|g(U_i^T V_j), \sigma_R^2)]^{I_{ij}^R} \tag{1}$$

where $N(x|\mu, \sigma^2)$ is the probability density function of the Gaussian distribution with mean μ and variance σ^2, and I_{ij}^R is the indicator function that is equal to 1 if user u_i rated item v_j and equal to 0 otherwise. The function $g(x)$ is the logistic function $g(x) = 1/(1 + exp(-x))$, which maps the value of predictions $(U_i^T V_j)$ within the range [0, 1]. We place zero-mean spherical Gaussian priors on user and item feature vectors similar to [1]:

$$P(U|\sigma_U^2) = \prod_{i=1}^{m} N(U_i|0, \sigma_U^2 I)$$
$$P(V|\sigma_V^2) = \prod_{j=1}^{n} N(V_j|0, \sigma_V^2 I) \tag{2}$$

Hence, through a Bayesian inference, we obtain:

$$\begin{aligned}
&P(U, V|R; \sigma_R^2, \sigma_U^2, \sigma_V^2) \propto P(R|U, V; \sigma_R^2)P(U|\sigma_U^2)P(V|\sigma_V^2)\\
&\quad= \prod_{i=0}^{m}\prod_{j=0}^{n}[N(r_{ij}|g(U_i^T V_j), \sigma_R^2)]^{I_{ij}^R}\\
&\quad\times \prod_{i=1}^{m} N(U_i|0, \sigma_U^2 I) \times \prod_{j=0}^{n} N(V_j|0, \sigma_V^2 I)
\end{aligned} \tag{3}$$

This basic matrix factorization model bases only on the user-item rating matrix, which we call the PMF model in this paper; it does not consider the information on users' social connections relationships. The corresponding graphical model is presented in Fig. 1.

3.2 Social Network Matrix Factorization

Here, suppose we have a directed social network graph $G = (u, \varepsilon)$,where $u = u_i$ represents all the users in the social network, and ε represents the social connections between users. Let $C = c_{ik}$ denote the $m \times m$ matrix of graph G,

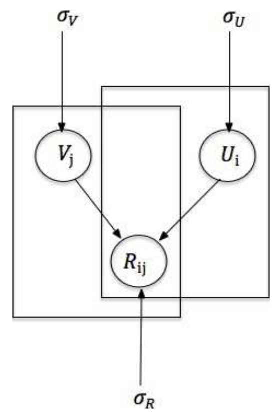

Fig. 1 Graphical model for PMF.

which is also called the social network matrix. For a pair of vertices, user u_i and user u_j, let $c_{ij} \in (0, 1]$ denote the two users' social relation weight.

We use the matrix factorization method to derive a high-quality l-dimensional latent user feature representation U by analyzing the matrix C. The latent user feature space is the same as the user feature space in the user-item rating matrix. Let $U \in R^{l\times m}$ and $Z \in R^{l\times m}$ be the user feature space and user factor feature space, respectively. Similarity, we obtain the conditional distribution over the observed users' social relationships:

$$P(C|U, Z, \sigma_C^2) = \prod_{i=0}^{m}\prod_{k=0}^{m}[N(C_{ik}|g(U_i^T Z_k), \sigma_C^2)]^{I_{ik}^C} \tag{4}$$

We also place zero-mean spherical Gaussian priors on latent user and factor feature vectors:

$$P(U|\sigma_U^2) = \prod_{i=1}^{m} N(U_i|0, \sigma_U^2 I)$$
$$P(Z|\sigma_Z^2) = \prod_{k=1}^{m} N(Z_k|0, \sigma_Z^2 I) \tag{5}$$

In order to reflect how user's social connections will affect this user's judgement on items, we fuse the user's social network matrix and user-item rating matrix into a consistent and compact feature representation. The posterior distribution for the recommendation is given by:

$$\begin{aligned}
&P(U, V, Z|R, C; \sigma_R^2, \sigma_C^2, \sigma_U^2, \sigma_V^2, \sigma_Z^2)\\
&\quad\propto P(R|U, V, \sigma_R^2)P(C|U, Z, \sigma_C^2)P(U|\sigma_U^2)P(V|\sigma_V^2)P(Z|\sigma_Z^2)\\
&\quad= \prod_{i=1}^{m}\prod_{j=1}^{n}[N(r_{ij}|g(U_i^T V_j); \sigma_R^2)]^{I_{ij}^R}\\
&\quad\times \prod_{i=1}^{m}\prod_{k=1}^{m}[N(c_{ik}|g(U_i^T Z_k), \sigma_C^2)]^{I_{ik}^C}\\
&\quad\times \prod_{i=1}^{m} N(U_i|0, \sigma_U^2 I) \times \prod_{j=1}^{n} N(V_j|0, \sigma_V^2 I)\\
&\quad\times \prod_{k=1}^{m} N(Z_k|0, \sigma_K^2 I)
\end{aligned} \tag{6}$$

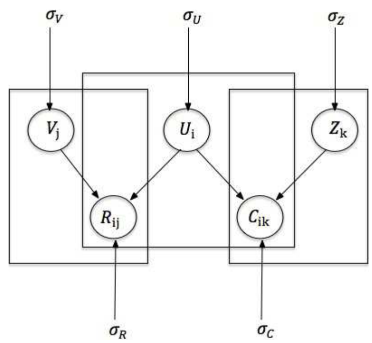

Fig. 2 Graphical model for SoPMF.

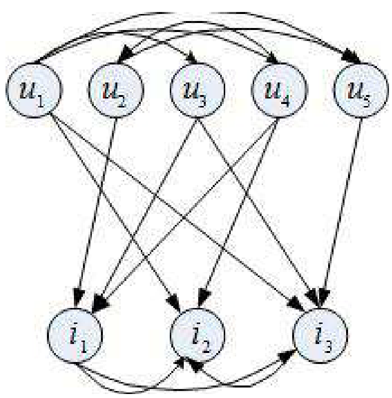

Fig. 3 Recommendation with users and items relationships.

This is the probabilistic matrix factorization based social recommendation (SoPMF). The graphical model corresponding to Eq. (6) is shown in Fig. 2. In the learning phase of this model, the user-item rating matrix and user's social network matrix are used to learn the user latent feature matrix and the item latent feature matrix, then the inner product of the column vectors U_i and V_j is used to predict the rating value of user u_i on item v_j.

3.3 Implicit Social Recommendation

As can be seen from the above, traditional social recommendation systems only consider the influence of social networks from the user's perspective and assume items are independent and identically distributed. However, the items relation information may be important because people select related items in different scenarios. Moreover, traditional social recommendation systems need explicit user social relationships to perform recommendations, but explicit user social relationships are not available in many recommendation systems.

In this section, an alternative algorithm, ISRec, is proposed. As shown in Fig. 3, we first compute the user social relationships information if explicit user social relationships are unavailable; meanwhile, we take the items relations into account when generating recommendation results.

The implicit user social relationships matrix can be constructed as follows:

First, we use the existing user-item rating matrix to compute the similarity between every user pair, and then use this similarity as an implicit social relationship between users. For the similarity, we only compute those users who have co-rated at least 10 items. In this way, we compute an implicit user social relationships matrix. Here, we denote the implicit user social relationships matrix as C, and now we can use social recommendation with this matrix when explicit user relationships are unavailable.

When computing the similarity between two users, we use the popular Pearson Correlation Coefficient approach (PCC), which is defined as:

$$s_{ij} = \frac{\sum_{k \in I(i) \bigcap I(f)} (r_{ik} - \bar{r}_i)(r_{fk} - \bar{r}_f)}{\sqrt{\sum_{k \in I(i) \bigcap I(f)} (r_{ik} - \bar{r}_i)^2} \cdot \sqrt{\sum_{k \in I(i) \bigcap I(j)} (r_{fk} - \bar{r}_f)^2}} \quad (7)$$

where $I(i)$ is a set of items rated by user u_i and $\bar{r}_i$ is the average rating of user u_i. From this definition, user similarity s_{if} ranges from [-1, 1], and a large value means user u_i and user u_f are more similar. Here, we use a function $f(x) = (x+1)/2$ to map the values of s_{ij} to the interval [0,1].

Motivated by the implicit user social relationships, we can similarly compute the items relationships matrix S in order to improve the item latent feature matrix V. Let $V \in R^{l \times n}$ and $W \in R^{l \times n}$ to be the latent item feature matrix and latent item factor feature matrix, respectively. The latent item feature space is the same as the item feature space in the user-item rating matrix. We define the conditional distribution over the observed connections of items as:

$$P(S|W, V; \sigma_S^2) = \prod_{t=1}^{n} \prod_{j=1}^{n} [N(S_{tj}|g(W_t^T V_j))]^{I_{tj}^S} \quad (8)$$

where I_{tj}^S is the indicator function that is equal to 1 if item v_t is connected to item v_j, 0 otherwise. We place zero-mean spherical Gaussian priors on latent item and item factor feature vectors:

$$P(V|\sigma_V^2) = \prod_{j=1}^{n} N(V_j|0, \sigma_V^2 I) \quad (9)$$

$$P(W|\sigma_W^2) = \prod_{t=1}^{n} N(W_t|0, \sigma_W^2 I) \quad (10)$$

In order to use the implicit social relationships of user and item relationships information, we model the problem using the graphical model shown in Fig. 4, which fuses the implicit user social relationships matrix, item relationships matrix, and user-item rating matrix together. Based on Fig. 4, the log of the posterior distribution is given by:

$$\begin{aligned} &ln\ p(U, V, Z, W|R, C, S; \sigma_R^2, \sigma_C^2, \sigma_S^2, \sigma_U^2, \sigma_V^2, \sigma_Z^2, \sigma_W^2) \\ &= -\frac{1}{2\sigma_R^2} \sum_{i=1}^{m} \sum_{j=1}^{n} I_{ij}^R (r_{ij} - g(U_i^T V_j))^2 \\ &\quad -\frac{1}{2\sigma_C^2} \sum_{i=1}^{m} \sum_{k=1}^{m} I_{ik}^C (c_{ik} - g(U_i^T Z_k))^2 \end{aligned}$$

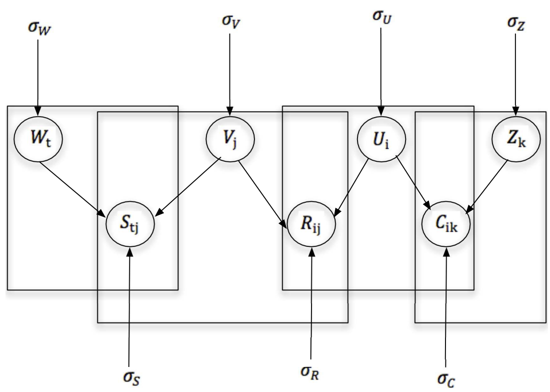

Fig. 4 Graphical model for ISRec.

$$
\begin{aligned}
&-\frac{1}{2\sigma_S^2}\sum_{t=1}^{n}\sum_{j=1}^{n} I_{tj}^{S}(s_{tj}-g(W_t^T V_j))\\
&-\frac{1}{2\sigma_U^2}\sum_{i=1}^{m} U_i^T U_i-\frac{1}{2\sigma_V^2}\sum_{j=1}^{n} V_j^T V_j\\
&-\frac{1}{2\sigma_Z^2}\sum_{k=1}^{m} Z_k^T Z_k-\frac{1}{2\sigma_W^2}\sum_{t=1}^{n} W_t^T W_t\\
&-\frac{1}{2}(\sum_{i=1}^{m}\sum_{j=1}^{n} I_{ij}^{R})ln\sigma_R^2-\frac{1}{2}(\sum_{i=1}^{m}\sum_{k=1}^{m} I_{ik}^{C})ln\sigma_C^2\\
&-\frac{1}{2}(\sum_{t=1}^{n}\sum_{j=1}^{n} I_{tj}^{S})ln\sigma_S^2-\frac{1}{2}(ml\ ln\sigma_U^2\\
&+nl\ ln\sigma_V^2+ml\ ln\sigma_Z^2+nl\ ln\sigma_S^2)+C \qquad (11)
\end{aligned}
$$

where C is a constant that does not depend on the parameters. Maximizing the log-posterior over four latent features with hyperparameters kept fixed is equivalent to minimizing the following sum-of-squared-errors objective function with quadratic regularization terms:

$$
\begin{aligned}
E(R,C,S,U,V,Z,W)&=\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{n} I_{ij}^{R}(r_{ij}-g(U_i^T V_j))^2\\
&+\frac{\lambda_C}{2}\sum_{i=1}^{m}\sum_{k=1}^{m} I_{ik}^{C}(c_{ik}-g(U_i^T Z_k))^2\\
&+\frac{\lambda_S}{2}\sum_{t=1}^{n}\sum_{j=1}^{n} I_{tj}^{S}(s_{tj}-g(W_t^T V_j))\\
&+\frac{\lambda_U}{2}\|U\|_F^2+\frac{\lambda_V}{2}\|V\|_F^2+\frac{\lambda_Z}{2}\|Z\|_F^2+\frac{\lambda_W}{2}\|W\|_F^2 \qquad (12)
\end{aligned}
$$

where $\lambda_C=\sigma_R^2/\sigma_C^2, \lambda_S=\sigma_R^2/\sigma_S^2, \lambda_U=\sigma_R^2/\sigma_U^2, \lambda_V=\sigma_R^2/\sigma_V^2$, $\lambda_Z=\sigma_R^2/\sigma_Z^2, \lambda_W=\sigma_R^2/\sigma_W^2$, and $\|\bullet\|_F^2$ denote the Frobenius norm. A local minimum of the objective function given by (12) can be found by performing gradient descent in W, Z, U, and V:

$$
\begin{aligned}
\frac{\partial E}{\partial U_i}&=\sum_{j=1}^{n} I_{ij}^{R} g'(U_i^T V_j)(g(U_i^T V_j)-r_{ij})V_j\\
&+\lambda_C\sum_{k=1}^{m} I_{ik}^{C} g'(U_i^T Z_k)(g(U_i^T Z_k)-c_{ik})Z_k+\lambda_U U_i \qquad (13)\\
\frac{\partial E}{\partial V_j}&=\sum_{i=1}^{m} I_{ij}^{R} g'(U_i^T V_j)(g(U_i^T V_j)-r_{ij})U_i\\
&+\lambda_S\sum_{t=1}^{n} I_{tj}^{S} g'(W_t^T V_j)(g(W_t^T V_j)-s_{tj})W_t+\lambda_V V_j \qquad (14)\\
\frac{\partial E}{\partial Z_k}&=\lambda_C\sum_{j=1}^{n} I_{ik}^{C} g'(U_i^T Z_t)(g(U_i^T Z_t)-c_{ik})U_i+\lambda_Z Z_k \qquad (15)\\
\frac{\partial E}{\partial W_t}&=\lambda_S\sum_{i=1}^{m} I_{tj}^{S} g'(W_t^T V_j)(g(W_t^T V_j)-s_{tj})V_j+\lambda_W W_t \qquad (16)
\end{aligned}
$$

where $g'(x)=exp(x)/(1+exp(x))^2$ is the derivative of the logistic function.

3.4 Complexity Analysis

In our proposed recommender system model, the main computation cost is in evaluating objection E and corresponding gradients on variables. Due to the sparsity of matrices R, C, and S, so the complexity of evaluating objection E is $O(\rho_R l+\rho_C l+\rho_S l)$, where ρ_R, ρ_C and ρ_S are the numbers of non-zero entries in matrices R, C and S, and l is the number of dimensions of latent feature space. The complexities of (13), (14), (15), (16) are $O(\rho_R l+\rho_C l)$, $\rho_R l+\rho_S l$, $\rho_C l$ and $\rho_S l$. Therefore, the total complexity for one iteration of the model is $O(\rho_R l+\rho_C l+\rho_S l)$, which means that the complexity of the algorithm is linear with the number of observations in the three sparse matrices. This demonstrates that our approach is efficient and is scalable to large datasets.

4. Experimental Analysis

In this section we conduct several experiments to give answers to the following questions:

(1) How effective is our approach compared to a basic matrix factorization method when explicit social information is unavailable?

(2) How effective is our approach compared to a social recommender system when explicit social information is available?

(3) How can the items relationship information help to improve the recommendation results, compared to taking the items independently?

4.1 Data Description and Metrics

We use the MovieLens 10M/100K dataset in our experiment to evaluate the algorithms when explicit social information is not available, and use the Epinions dataset to evaluate the performance of the algorithms when it is available, because this latter dataset has a trust relation between users, in addition to the user-item rating matrix.

The MovieLens 10M/100K is a relatively small dataset contains 1,000,000 user-item ratings (scaled from 1 to 5), rated by 943 users on 1642 items. The Epinions dataset contains 49,290 users, who rated a total of 139,738 different items at least once, and it also has 487,181 issued trust statements which we can use as explicit user social network information.

The main purpose of recommender systems is to predict users' future likes and interests. Multiple metrics exist to measure various aspects of recommendation performance. Two notable metrics, Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE), are used to measure the closeness of predicted ratings to the true ratings. MAE is defined as

$$MAE = \frac{\sum_{i,j} |r_{ij} - \hat{r}_{ij}|}{N} \tag{17}$$

where r_{ij} denotes the rating user u_i gives to item v_j, $\hat{r}_{ij}$ denotes the related predicted rating, and N denotes the number of tested ratings. RMSE is defined as:

$$RMSE = \sqrt{\frac{\sum_{i,j} (r_{ij} - \hat{r}_{ij})^2}{N}} \tag{18}$$

Lower MAE and RMSE results correspond to higher prediction accuracy. Since RMSE squares the error before summing it, it tends to penalize large errors more heavily. As these metrics treat all ratings equally, no matter what their positions are in the recommendation list, they are not optimal for some common tasks such as finding a small number of objects that are likely to be appreciated by a given user (finding good objects). Yet, due to their simplicity, RMSE and MAE are widely used in the evaluation of recommender systems.

4.2 Performance Analysis

In this section, the performance improvement of our Implicit Social Recommendation (ISRec) algorithm is shown, and the algorithms compared in this paper are as follows:

(1) PMF: Probabilistic Matrix Factorization (PMF) method. The details of this method are introduced in Sect. (3.1).

(2) SoPMF: social recommender method that uses explicit user social information. The details of this method are introduced in Sect. (3.2).

(3) ISRec: the method which is proposed in this paper which uses the whole implicit user and item social information.

(4) IuSRec: this is the Implicit Social Recommendation method which only uses the whole implicit user social information.

(5) IuSRe$c^{0.75}$: IuSRec method using implicit user social information with a value greater than 0.75.

(6) ISRe$c^{0.75}$: ISRec method using implicit social information with a value greater than 0.75.

In order to compare every method fairly,we use similar parameter settings for those common parameters adopted in

Table 1 MovieLens DataSet `RMSE` comparison.

Training Data	PMF	$IuSRec^{0.75}$	$ISRec^{0.75}$	IuSRec	ISRec
99%	0.6807	0.6653	0.6636	0.6103	0.6091
80%	0.6872	0.6731	0.6713	0.6216	0.6197
50%	0.6892	0.6775	0.6760	0.6379	0.6343
20%	0.6902	0.6810	0.6794	0.6562	0.6503
10%	0.6997	0.6813	0.6796	0.6614	0.6548

Table 2 MovieLens DataSet `MAE` comparison.

Training Data	PMF	$IuSRec^{0.75}$	$ISRec^{0.75}$	IuSRec	ISRec
99%	0.6206	0.6061	0.6044	0.5521	0.5507
80%	0.6207	0.6141	0.6124	0.5640	0.5619
50%	0.6274	0.6187	0.6172	0.5807	0.5768
20%	0.6298	0.6218	0.6201	0.5980	0.5918
10%	0.6300	0.6219	0.6202	0.6027	0.5958

Table 3 Epinions DataSet `RMSE` comparison.

Training Data	PMF	$SoPMF$	$ISRec^{0.75}$	ISRec
90%	0.8038	0.7515	0.8007	0.797
10%	0.8040	0.7707	0.8015	0.7983

Table 4 Epinions DataSet `MAE` comparison.

Training Data	PMF	$SoPMF$	$ISRec^{0.75}$	ISRec
90%	0.7452	0.6928	0.7421	0.7381
10%	0.7455	0.7132	0.7430	0.7394

all the approaches. In this paper, for PMF, SoPMF, IuSRec, IuSRe$c^{0.75}$, ISRec, ISRe$c^{0.75}$, the dimension of latent feature vectors is set to 50, and the maximum iteration time is set to 100. At the same time, we use the setting $\lambda_C = 10, \lambda_S = 1, \lambda_U = \lambda_V = \lambda_W = \lambda_Z = 0.01$.

MovieLens Dataset: We use different amounts of training data (99%, 80%, 50%, 20%, 10%) to test all the algorithms. Training data 99% means that we randomly select 99% of the ratings from the MovieLens dataset as the training data to predict the remaining 1% of rating. From Table 1 and 2, we can see that: first, our approach significantly outperforms the traditional methods which only use a user-item rating matrix; second, when comparing the methods ISRec and IuSRec, we can see that the items relations have a great impact on recommendation accuracy; third, the results of $ISRec^0.75$ and ISRec show that taking the whole information as our compute method in this paper performs better than just using part of that information.

Epinions Dataset: On this dataset, we compare the implicit social recommendation with the explicit social information, and the results from Table 3 and 4 show that our approach performs only slightly worse than social recommendation with explicit social information, but this is still better than PMF, which does not use the social information.

One task we target in this paper is to provide accurate recommendations when users only supply a few ratings. Hence, we evaluate how different methods perform on different users based on how many ratings the users rated in the training datasets. We first group all the users based on the number of observed ratings in the training data (we

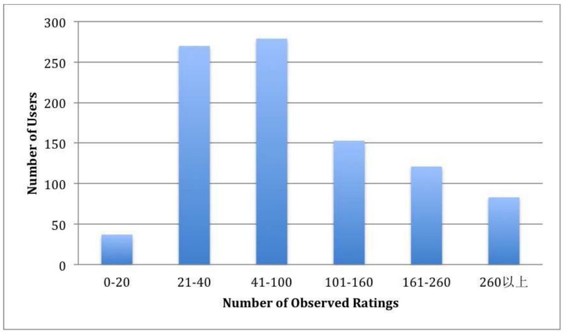

Fig. 5 Distribution of training data (80% as training data).

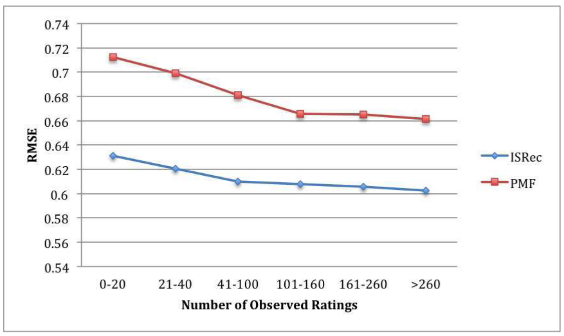

Fig. 6 RMSE performance comparison on different user rating scales (80% as training data).

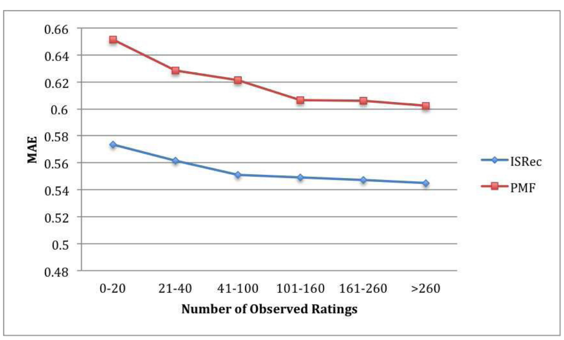

Fig. 7 MAE performance comparison on different user rating scales (80% as training data).

choice 80% MovieLens dataset), and then evaluate prediction accuracies of different user groups. In order to interpret the results more intuitively, we include the baseline method PMF for comparison since it does not include any social information. We also test a number of other training dataset, including 10%, 20%, 50% and 99% as training dataset. For all of them, we observe similar trends in the analysis, and the results are not qualitatively different. Hence, we only report the results using 80% MovieLens dataset as training set. The experimental results are shown in Fig. 5, 6 and 7.

Users are grouped into 6 classes: "[0, 20]", "[21, 40]", "[41, 100]", "[101, 160]", "[161, 260]" and "> 260". Figure 5 summarizes the distributions of testing data according to groups in the training data. From Fig. 6 and Fig. 7 we

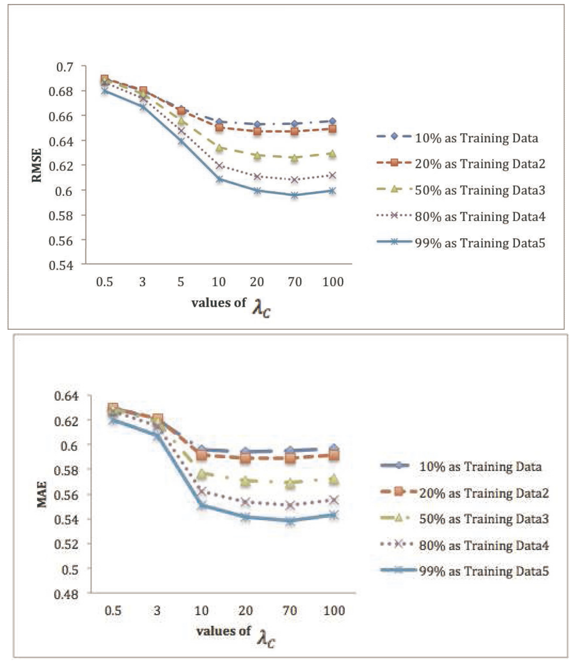

Fig. 8 Impact of parameter λ_C.

can see that the method ISRec with implicit social information consistently outperforms the PMF method in all the user groups. We also notice that our method performs much better than PMF when users have few ratings. The RMSE improved by 11.4%, and the MAE improved by 11.9% for user rating numbers between 0 and 20. For user rating numbers more than 260, RMSE improved by 9% and MAE by 9.5%. This show that the implicit social information play more important role in making recommendations when users supply a few ratings and the proposed method alleviated the data sparsity problem in the recommender systems.

4.3 Impact of Parameter λ_C

In our model, the parameter λ_C balances the information from user implicit information and the items' implicit information. It controls the impact of the user implicit social information matrix. If we set λ_C=0, it means we do not use the implicit social information from users. If we set λ_C to +inf, it means we only utilize users' implicit social information.

We test the impact of the parameter λ_C when we set $\lambda_U = \lambda_V = \lambda_Z = \lambda_W$=0.005, λ_S=1. Figure 8 shows the impact of λ_C on RMSE. From the results we can see that the value of λ_C impacts the prediction results significantly. As the value of λ_C increases, the values of RMSE decrease at first, but when λ_C surpasses a certain threshold, the values of RMSE increase as the value of λ_C increases. The results meet the intuition that purely using the users' implicit social information or user-item rating information for predictions cannot get better results than fusing this information together.

5. Conclusions

We compute implicit social information among users and items and propose an implicit social recommendation algorithm in order to estimate social recommendations on a dataset where explicit user social information is unavailable. The experimental results show that the performance of our ISRec approach is poor compared to when explicit user social information is used; however, where this information is not available, our method can significantly improve prediction accuracies. Thus, we can use ISRec for most real word recommendation systems, especially when the user's explicit social connection information is unavailable.

Acknowledgements

This work is supported by the National Key project of Scientific and Technical Supporting Programs of China (Grant No. 2012BAH01F02, 2013BAH10F01, 2013BAH07F02); the National Natural Science Foundation of China (Grant No.61072060); the National High Technology Research and Development Program of China Grant No. 2011AA100706); the Research Fund for the Doctoral Program of Higher Education (Grant No. 20110005120007); the Fundamental Research Funds for the Central Universities; Engineering Research Center of Information Networks, Ministry of Education.

References

[1] A. Mnih and R. Salakhutdinov, "Probabilistic matrix factorization," Advances in Neural Information Processing Systems, pp.1257–1264, 2007.
[2] R. Jin, J.Y. Chai, and L. Si, "An automatic weighting scheme for collaborative filtering," Proc. 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp.337–344, 2004.
[3] G. Linden, B. Smith, and J. York, "Amazon. com recommendations: Item-to-item collaborative filtering," IEEE Internet Computing, vol.7, no.1, pp.76–80, 2003.
[4] H. Ma, I. King, and M.R. Lyu, "Effective missing data prediction for collaborative filtering," Proc. 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp.39–46, 2007.
[5] J. Wang, A.P. De Vries, and M.J.T. Reinders, "Unifying user-based and item-based collaborative filtering approaches by similarity fusion," Proc. 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp.501–508, 2006.
[6] T. Hofmann, "Collaborative filtering via gaussian probabilistic latent semantic analysis," Proc. 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval, pp.259–266, 2003.
[7] T. Hofmann, "Latent semantic models for collaborative filtering," ACM Trans. Information Systems (TOIS), vol.22, no.1, pp.89–115, 2004.
[8] L. Si and R. Jin, "Flexible mixture model for collaborative filtering," ICML, vol.3, pp.704–711, 2003.
[9] H. Ma, H. Yang, M.R. Lyu, et al., "Sorec: Social recommendation using probabilistic matrix factorization," Proceedings of the 17th ACM conference on Information and knowledge management, pp.931–940, 2008.
[10] H. Ma, M.R. Lyu, and I. King, "Learning to recommend with trust and distrust relationships," Proc. Third ACM Conference on Recommender Systems, pp.189–196, 2009.
[11] M. Jamali and M. Ester, "A matrix factorization technique with trust propagation for recommendation in social networks," Proc. Fourth ACM Conference on Recommender Systems, pp.135–142, 2010.
[12] Y. Shen and R. Jin, "Learning personal+social latent factor model for social recommendation," Proc. 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp.1303–1311, New York, 2012.
[13] P. Resnick, N. Iacovou, M. Suchak, et al., "GroupLens: An open architecture for collaborative filtering of netnews," Proc. 1994 ACM Conference on Computer Supported Cooperative Work, pp.175–186, 1994.
[14] J.S. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," Proc. Fourteenth Conference on Uncertainty in Artificial Intelligence, pp.43–52, 1998.
[15] L. Wu, E. Chen, Q. Liu, et al., "Leveraging tagging for neighborhood-aware probabilistic matrix factorization," Proc. 21st ACM International Conference on Information and Knowledge Management, pp.1854–1858, 2012.
[16] H. Ma, "An experimental study on implicit social recommendation," Proc. 36th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp.73–82, 2013.
[17] J.D.M. Rennie and N. Srebro, "Fast maximum margin matrix factorization for collaborative prediction," Proc. 22nd International Conference on Machine Learning, pp.713–719, 2005.
[18] R. Salakhutdinov and A. Mnih, "Bayesian probabilistic matrix factorization using Markov chain Monte Carlo," Proc. 25th International Conference on Machine Learning, pp.880–887, 2008.
[19] B. Sarwar, G. Karypis, J. Konstan, et al., "Item-based collaborative filtering recommendation algorithms," Proc. 10th International Conference on World Wide Web, pp.285–295, 2001.
[20] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," IEEE Trans. Knowl. Data Eng., vol.17, no.6, pp.734–749, 2005.
[21] H. Yildirim and M.S. Krishnamoorthy, "A random walk method for alleviating the sparsity problem in collaborative filtering," Proc. 2008 ACM Conference on Recommender Systems, pp.131–138, 2008.
[22] T.C. Zhou, H. Ma, I. King, et al., "Tagrec: Leveraging tagging wisdom for recommendation," International Conference on Computational Science and Engineering, 2009. CSE'09, vol.4, pp.194–199, 2009.
[23] F. Liu and H.J. Lee, "Use of social network information to enhance collaborative filtering performance," Expert Systems with Applications, vol.37, no.7, pp.4772–4778, 2010.
[24] H. Ma, D. Zhou, C. Liu, et al., "Recommender systems with social regularization," Proc. Fourth ACM International Conference on Web Search and Data Mining, pp.287–296, 2011.
[25] S.H. Yang, B. Long, A. Smola, et al., "Like like alike: Joint friendship and interest propagation in social networks," Proc. 20th International Conference on World Wide Web, pp.537–546, 2011.

Yusheng Li was born in 1983, he is a Ph.D. candidate in PCN&CAD Research Center, Beijing University of Posts and Telecommunications, Beijing 100876, China. His major research interests include machine learning and recommender systems. He is a member of ACM and IEICE.

Meina Song was born in 1974, she is a Ph.D., Professor at the Department of Computer Science and Technology of Beijing University of Posts and Telecommunication. Her research interests include service methodology, service system architecture, cloud compute and big data.

Haihong E was born in 1982, she is a Ph.D., Lecturer at the Department of Computer Science and Technology of Beijing University of Posts and Telecommunication. Her research interests include personalized recommendation system, service network, and trusted service.