Social Recommendation Using Low-Rank Semidefinite Program

Jianke Zhu[1], **Hao Ma**[2], **Chun Chen**[1] **and Jiajun Bu**[1]
[1]Zhejiang Provincial Key Laboratory of Service Robot, College of Computer Science, Zhejiang University
[2]Microsoft Research
{jkzhu, chenc, bjj}@zju.edu.cn, haoma@microsoft.com

Abstract

The most critical challenge for the recommendation system is to achieve the high prediction quality on the large scale sparse data contributed by the users. In this paper, we present a novel approach to the social recommendation problem, which takes the advantage of the graph Laplacian regularization to capture the underlying social relationship among the users. Differently from the previous approaches, that are based on the conventional gradient descent optimization, we formulate the presented graph Laplacian regularized social recommendation problem into a low-rank semidefinite program, which is able to be efficiently solved by the quasi-Newton algorithm. We have conducted the empirical evaluation on a large scale dataset of high sparsity, the promising experimental results show that our method is very effective and efficient for the social recommendation task.

1 Introduction

As the rapid growth of Web 2.0 sites and Web applications, the recommender systems have become more and more important for the end users to filter out the large amount of useless information. Currently, the recommendation is not only of the great research interest to be explored, but also of the huge commercial opportunities in practice. During the past decade, the extensive research efforts in information retrieval (Linden, Smith, and York 2003; Hofmann 2003), machine learning (Breese, Heckerman, and Kadie 1998; Zhang and Koren 2007; Si and Jin 2003) and data mining domains have devoted to the recommendation related techniques. More importantly, the recommender systems have already been deployed in the production recommendation at Amazon and the music recommendation at iTunes.

The recommender systems are mainly based on the technique of collaborative filtering (Breese, Heckerman, and Kadie 1998), which automatically predicts the interest of an active user using the information collected from the other users. In general, the users only rate a very small portion of the whole item set. Therefore, the grand challenge for the recommender system is how to deal with the large scale dataset with lots of missing entries. Moreover, the conventional recommender systems are solely dependent on the information of user-item ratings. Obviously, they ignore the important clues of the social connections or the trust relations among the users, which may lead to the undesired prediction performance for the recommender system with a very large number of users.

Recently, there are a surge of research interests on the social recommendation (Ma et al. 2008; Ma, King, and Lyu 2009; Jamali and Ester 2010), which tend to employ the user connections to improve the prediction quality. These methods are dependent on the probabilistic matrix factorization with the gradient descent optimization. However, the shared low-dimensional latent subspace in (Ma et al. 2008) cannot effectively capture the underlying social connections and the tastes among the users. Although the ensemble method (Ma, King, and Lyu 2009) alleviates this problem by directly fusing the users' favors and their trusted friends' decisions, it involves with the intensive computation on calculating the fused prediction value. As the total number of user-item ratings is usually quite large, the training process may become time-consuming in case of the users with many social connections. Additionally, a weighting parameter has to be set empirically in order to balance the tastes of the users' and their friends'.

Since the existing approaches have the limitations, there is a need for developing the new techniques to resolve these challenges. In this paper, we address the current issues by introducing a low-rank semidefinite program (LRSDP) approach. LRSDP (Burer and Monteiro 2003; Burer and Choi 2006) is developed to efficiently solve the rank constrained semidefinite program (SDP), which has the advantages of the low computational complexity and memory requirement. Therefore, it is capable of tackling the critical challenge of social recommendation with the large scale sparse data and the extra constraints. In contrast to the previous social recommendation approaches, we directly regularize the user-specific latent space through the graph Laplacian (Cvetkovic, Doob, and Sachs 1998), which is able to capture the underlying social relationships between the different users. Motivated by the recent work (Mitra, Sheorey, and Chellappa 2010) in the computer vision domain, we formulate the social recommendation problem into a low-rank factorization problem with the graph Laplacian regularization, which can be very efficiently solved by an LRSDP optimization algorithm (Burer and Monteiro 2003). The experimental evaluations on the Epinions dataset show the encouraging results of the proposed method comparing to the state-of-the-art approaches in terms of both prediction accuracy and time complexity.

The rest of this paper is organized as follows. Section 2 goes somewhat deeper into the previous methods for the recommender system. Section 3 presents the formulation of recommendation using low-rank matrix factorization. Section 4 proposes our novel approach to tackling the social recommendation with the graph Laplacian, and presents the LRSDP optimization scheme. Section 5 describes the details of our experimental results. Section 6 sets out our conclusion and addresses some future work.

2 Related Work

The recommender systems have already received quite a bit of attention, which mainly employ the techniques on collaborative filtering. Moreover, the conventional collaborative filtering approaches are either based on the neighborhood assumption or dependent on a learning model. For the neighborhood-based methods (Linden, Smith, and York 2003; Breese, Heckerman, and Kadie 1998; Jin, Chai, and Si 2004), they mainly try to search for the similar active users or items in order to predict the ratings. On the other hand, the model-based approaches (Hofmann 2003; Zhang and Koren 2007; Si and Jin 2003) aim to take advantage of the statistical and machine learning techniques to train a compact model from the rating data. Additionally, the recommendation is viewed as a matrix complete problem, which can be formulated as a semidefinite program (Srebro, Rennie, and Jaakkola 2005).

Our work is closely related to the low-rank matrix factorization method for the recommender system, which makes the assumption that only a very small number of latent factors affect the users' preferences in the user-item rating matrix. Moreover, a user's preference vector could be represented by a linear combination of these latent factors. Assuming that the entries in user-item matrix are corrupted by Gaussian noise, the low-rank matrix could be approximated by Singular Value Decomposition (SVD). Although the SVD approaches enjoy the merits of the closed-form solution, it is not very effective to deal with the large-scale rating matrix having a large amount of missing entries. In addition, the probabilistic matrix factorization method (Salakhutdinov and Mnih 2008) is proposed to take advantages of the latent semantic model and the iterative gradient descent optimization. Most recently, the matrix factorization with missing data problem is formulated into a low-rank semidefinite program, which demonstrates the promising results on the computer vision task such as structure from motion (Mitra, Sheorey, and Chellappa 2010).

Note that all the above methods are based on the assumption that the users are independent, and ignores the social connections among the different users. To solve this issue, several remedies are proposed to incorporate the social information into the recommender systems. (Ma et al. 2008) developed a factor analysis method using the probabilistic graphical model which combines the user-item matrix with users' social trust networks by sharing a common latent low-dimensional user-feature subspace. Due to the lack of the physical meanings, the sharing approach does not reveal the underlying relations among the users. In (Ma, King, and Lyu 2009), an ensemble probabilistic matrix factorization method is presented to carefully take consideration of the tastes among the users, which demonstrates the state-of-art performance on the large scale dataset with a large amount of missing entries. However, directly fusing the social information into the user-item matrix increases the computational cost during the optimization, especially for the dataset with a large number of user-trust connections.

Unlike the existing approaches, our presented method leverages the matrix factorization with a low-rank semidefinite program, which is based on the quasi-Newton method having the low computational complexity. The key of our method is to impose the graph Laplacian regularization on the user-specific latent space to find the optimal latent factor matrices with the social constraints, which is further formulated into the LRSDP problem without incurring the heavy computational cost.

3 Recommendation by Matrix Factorization

Currently, a very effective approach to recommendation problem is to predict the missing data entries through factorizing the user-item rating matrix into the user-specific and item-specific matrices (Salakhutdinov and Mnih 2008; Ma et al. 2008; Ma, King, and Lyu 2009). The premise behind a low-dimensional factor model is that there is only a small number of latent factors influencing the preferences, and that a user's preference vector is dependent on how each factor applies to that user. Therefore, we aim to predict the users' rating value using the product of these recovered latent factors.

Given m users and n items in a recommender system, the objective of low-rank matrix factorization is to factorize the user rating matrix $M \in \mathbf{R}^{m\times n}$ into the d-rank user-specific latent factor matrix $U \in \mathbf{R}^{m\times d}$ and item-specific latent factor matrix $V \in \mathbf{R}^{n\times d}$ respectively. As there are usually a large number of items in the recommender system, and a user may only rate a small portion of the whole item set practically. As a result, the user rating matrix M is generally very sparse. It is important to note that the rank d is a quite small number comparing to the total number of the users m and the items n. To take into account of the large number of missing elements in M, we usually solve the following constrained minimization problem (Salakhutdinov and Mnih 2008; Ma et al. 2008):

$$\min_{U,V} \|I \otimes \left(M - U^\top V\right)\|_F^2 + \lambda\left(\|U\|_F^2 + \|V\|_F^2\right) \quad (1)$$

where $\|\cdot\|_F^2$ denotes the Frobenius norm, and $\otimes$ represents the Hadamard element-wise product. I is an indicator matrix with ones for the user-rated items and zeros for the missing data, and $\|U\|_F^2$ and $\|V\|_F^2$ are the regularization terms in order to avoid the overfitting issue. The regularization coefficient λ is employed to balance the weight between the data penalty term and the regularization term, which is usually set empirically.

Generally speaking, most of the recommender systems employ integer values from one to $\mathbf{r}_{\max}$ to represent the user's ratings on the items. Without loss of generality, we first transform the ratings in the matrix M into the interval $(0, 1)$ using the mapping function $g(x)$ as follows:

$$g(x) = \frac{x - \bar{\mathbf{r}}}{\mathbf{r}_{\max}} \tag{2}$$

where $\bar{\mathbf{r}}$ denotes the mean of $[1, \ldots, \mathbf{r}_{\max}]$. Then, the logistic function $\rho(x)$ is employed to bound each element of matrix multiplication $U^\top V$ within the range $(0, 1)$: $\rho(x) = \frac{1}{1+\exp(-x)}$. We abuse the notation by denoting the element-wise logistic function of a matrix as $\rho(A)$, and rewrite the minimization problem in Eqn. 1 into the following form:

$$\min_{U,V} \|I \otimes \left(M' - \rho\left(U^\top V\right)\right)\|_F^2 + \lambda\left(\|U\|_F^2 + \|V\|_F^2\right) \tag{3}$$

where M' denotes the users' rating value transformed by the mapping function $g(x)$ in Eqn. 2.

The optimization problem in Eqn. 3 aims to minimize the sum of reconstruction errors with the two quadratic regularization terms, which is usually solved by the gradient descent method (Salakhutdinov and Mnih 2008) or the alternative projection algorithm (Li and Yeung 2009).

4 Social Recommendation by LRSDP

In this section, we present the proposed low-rank semidefinite program approach to social recommendation task. We first give our formulation of social recommendation with the graph Laplacian regularization, and then solve the minimization problem by the low-rank semidefinite program optimization.

4.1 Low-Rank Matrix Factorization with Social Regularization

Before making a decision on purchasing a new product, we are used to obtain the valuable recommendations by consulting our friends with the fruitful experiences. To incorporate the social relationship information into the recommender system, we consider to penalize the differences between the user-specific latent feature vectors when there exist the direct connections between them.

There are several choices to impose the penalty on the user-specific feature space. In this paper, we propose to directly regularize the user-specific latent space through the graph Laplacian (Cvetkovic, Doob, and Sachs 1998), which has the solid theoretical foundation with many successful applications on manifold learning. Furthermore, the graph Laplacian can be viewed as a kind of discrete Laplace operator, which is capable of capturing the underlying relationships between the similar users.

Specifically, a graph Laplacian is defined as

$$L = \mathrm{diag}(S\mathbf{1}) - S$$

where $\mathbf{1}$ denotes a vector with all one elements. Moreover, $S \in R^{m\times m}$ is a similarity matrix. Let u_i and u_j denote the two users having the social connections, and then each element $S_{i,j}$ is calculated by:

$$S_{ij} = S_{ji} = \begin{cases} \phi(u_i, u_j), & u_i \text{ and } u_j \text{ are friends,} \\ 0, & \text{otherwise,} \end{cases}$$

where $\phi(u_i, u_j)$ represents the kernel function with respect to the two connected users. Moreover, there are three different kernel functions: namely, binary kernel, distance kernel and heat kernel.

For the binary kernel, $\phi(u_i, u_j)$ is simply set to one, which means that each friend contributes equally in the recommendation process. In the case of distance kernel, we try to find a similarity function which measures the difference of the two user u_i and u_j' taste. To this end, we can take advantages of the observations in the user rating matrix M', and calculate the distance between the two corresponding vectors.

Since the cosine similarity measure $\triangle_{\cos}$ yields better results in the empirical evaluation, it is chosen to measure the users' taste. In this paper, $\triangle_{\cos}$ is defined as: $\triangle_{\cos} = \frac{\mathbf{u}_i^\top \mathbf{u}_j}{\|\mathbf{u}_i\|\cdot\|\mathbf{u}_j\|}$,where $\mathbf{u}_i$ and $\mathbf{u}_j$ are the two rows in M'. Due to the large number of missing elements in M', it is meaningful to measure the users' taste by considering the ratings of the same items. Therefore, we only calculate the cosine distance on the subset of items which both users have rated, which is equivalent to the vector space similarity defined in (Breese, Heckerman, and Kadie 1998). Furthermore, the heat kernel can be built by $\phi(u_i, u_j) = e^{-\frac{1-\triangle_{\cos}}{2\varsigma^2}}$, where ς denotes the width for a heat kernel, which is set to one empirically.

As suggested in (Li and Yeung 2009), the graph Laplacian is able to be integrated into the matrix factorization by adding an extra regularization term $\mathbf{tr}(U^\top LU)$ into the minimization problem in Eqn. 3:

$$\begin{aligned} \min_{U,V} \quad & \|I \otimes \left(M' - \rho\left(U^\top V\right)\right)\|_F^2 + \lambda\left(\|U\|_F^2 + \|V\|_F^2\right) \\ & +\eta\mathbf{tr}(U^\top LU) \end{aligned} \tag{4}$$

where η is the regularization coefficient for the Laplacian regularization, which can effectively penalize the deviations of latent user models from each other if they are linked in the social network.

Let $\mathbf{tr}(\cdot)$ denote the trace of a matrix, we reformulate the regularization term $\|U\|_F^2 + \|V\|_F^2$ into the trace form $\mathbf{tr}(U^\top U) + \mathbf{tr}(V^\top V)$. Thus, the above equation can be further written as follows:

$$\begin{aligned} \min_{U,V} \quad & \|I \otimes \left(M' - \rho\left(U^\top V\right)\right)\|_F^2 + \lambda\mathbf{tr}(V^\top V) \\ & +\mathbf{tr}\left(U^\top(\lambda I + \eta L)U\right) \end{aligned} \tag{5}$$

It can be clearly observed that the social information is naturally encoded into the regularization for the user-specific latent factors. More importantly, the above formulation for the graph Laplacian regularized low-rank matrix factorization is very similar to the optimization problem defined in Section 3.

4.2 LRSDP Optimization

Differently from the previous methods, that are based on either gradient descent optimization or the alternative projection algorithm, we propose a low-rank semidefinite programming (LRSDP) approach to the matrix factorization with graph Laplacian regularization. LRSDP (Burer and Monteiro 2003; Burer and Choi 2006) is essentially a rank constrained semidefinite programing (SDP), which is based on a quasi-Newton algorithm having the lower computational complexity and memory requirement than that of Newton's method. It is worthy of mentioning that the graph Laplacian constraints presented in Section 4.1 can be easily integrated into the LRSDP optimization framework.

As in (Boyd and Vandenberghe 2004), a standard SDP optimization has a set of linear equality constraints, and a matrix non-negativity constraint on an $n \times n$ matrix variable X, which is defined as follows:

$$\begin{aligned} \min \quad & \mathbf{tr}\left(C^\top X\right) \\ \text{s. t.} \quad & \mathbf{tr}\left(A_i^\top X\right) = \mathbf{b}_i, \quad i = 1, \dots, p \\ & X \succeq \mathbf{0} \end{aligned} \tag{6}$$

where C and $A_1, \dots, A_p$ are $n \times n$ real symmetric matrices, and $\mathbf{b}$ is a p dimensional vector. Note that the objective function is the form of a general real-valued linear function in the $n \times n$ symmetric matrix space:

$$\mathbf{tr}\left(C^\top X\right) = \sum_{i,j=1}^{n} C_{i,j} X_{ij}$$

In (Mitra, Sheorey, and Chellappa 2010), a relaxation approach is proposed to solve the low-rank matrix factorization problem by changing the variables $X = GG^\top$. $G \in \mathbf{R}^{(m+n)\times d}$ is a real matrix which stacks the variables U and V as follows:

$$G = \begin{bmatrix} U \\ V \end{bmatrix}$$

Therefore,

$$GG^\top = \begin{bmatrix} UU^\top & UV^\top \\ VU^\top & VV^\top \end{bmatrix}$$

As discussed in (Mitra, Sheorey, and Chellappa 2010), the regularization term $\|U\|_F^2 + \|V\|_F^2$ is equivalent to $\mathbf{tr}\left(GG^\top\right)$. Consequently, the minimization of matrix factorization in Eqn. 1 can be formulated into the following LRSDP problem:

$$\begin{aligned} \min_G \quad & \mathbf{tr}\left(GG^\top\right) \\ \text{s. t.} \quad & \mathbf{tr}\left(A_k^\top GG^\top\right) = M'_{i,j}, \quad k = 1, \dots, p \end{aligned} \tag{7}$$

where A_k are sparse matrices with the non-zero entries at indices $(i, j+m)$ and $(j+m, i)$ equal to 0.5, and p is the total number of users' ratings in M'.

In contrast to the original matrix factorization problem defined in Eqn. 1, we need take the extra considerations of the logistic function and the graph Laplacian regularization for the proposed social recommendation approach. In the following, we present the remedies to solve these issues.

As for the matrix factorization with the logistic function $\rho(x)$, we find that the minimization problem in Eqn. 3 is equivalent to:

$$\begin{aligned} \min_G \quad & \mathbf{tr}\left(GG^\top\right) \\ \text{s. t.} \quad & \mathbf{tr}\left(A_k^\top \rho\left(GG^\top\right)\right) = M'_{i,j}, k = 1, \dots, p \end{aligned} \tag{8}$$

Since all the rating values are mapped into the interval $(0, 1)$, we can utilize the inverse of logistic function and rewrite the equality constraints in Eqn. 8 as follows:

$$\mathbf{tr}\left(A_k^\top GG^\top\right) = \log \frac{M'_{ij}}{1 - M'_{ij}}, \quad k = 1, \dots, p \tag{9}$$

To integrate the graph Laplacian regularization into the LRSDP optimization framework, we take advantage of the property for matrix trace $\mathbf{tr}(ABC) = \mathbf{tr}(BCA)$. Thus, the user-specific regularization term in Equ. 3 is reformulated into the following form:

$$\mathbf{tr}\left(U^\top(\lambda\mathbf{I} + \eta L)U\right) = \mathbf{tr}\left((\lambda I + \eta L)UU^\top\right)$$

Then, we define an $(m+n) \times (m+n)$ symmetric matrix C as below:

$$C = \begin{bmatrix} \lambda I_{(m\times m)} + \eta L & \mathbf{0} \\ \mathbf{0} & I_{(n\times n)} \end{bmatrix}$$

Finally, we can write the low-rank matrix factorization with the graph Laplacian regularization into the following LRSDP problem:

$$\begin{aligned} \min_G \quad & \mathbf{tr}\left(C^\top GG^\top\right) \\ \text{s. t.} \quad & \mathbf{tr}\left(A_k^\top GG^\top\right) = \log \frac{M'_{ij}}{1 - M'_{ij}}, \\ & k = 1, \dots, p \end{aligned} \tag{10}$$

In this paper, the minimization problem in Eqn. 10 is solved by the augmented Lagrangian method proposed in (Burer and Monteiro 2003; Burer and Choi 2006), which is a nonlinear method using BFGS optimization.

5 Experimental Results

In this section, we present the details of our experimental implementation and report the empirical results on the social recommendation task.

5.1 Experimental Setup

We evaluate the performance of our presented LRSDP with graph Laplacian regularization approach on the Epinions dataset and compare it against several algorithms, such as Probabilistic Matrix Factorization (PMF), SoRec (Ma et al. 2008), recommendation purely by the trusted friends (Trust) and Social Trust Ensemble (STE) in (Ma, King, and Lyu 2009). Note that the alternative projection algorithm (Li and Yeung 2009) was designed for factorizing the dense matrix, which cannot be directly applied to the social recommendation problem with large scale sparse data.

In this paper, the Epinions dataset is employed as the testbed for the empirical evaluation on recommendation with social relation information. Epinions.com is a well-known review website which is established for sharing the knowledge on products. At Epinions, the visitors can utilize the reviews on a variety of items in order to assist hem on making the purchase decisions. They first register an account for free, and then begin submitting reviews according to their own personal opinions that may earn some reward and recognition. To post a review, users need to rate

Table 1: Statistics of User-Item Rating Matrix of Epinions

Statistics	User	Item
Max. Num. of Ratings	1960	7082
Avg. Num. of Ratings	12.21	7.56

Table 2: Statistics of Social Trust Network of Epinions

Statistics	Trust per User	Be Trusted per User
Max. Num.	1763	2443
Avg. Num.	9.91	9.91

the product or service with an integer rating value from one to five. Each member of Epinions maintains a trust list which presents a network of trust relationships among users.

The Epinions dataset consists of 51,670 users who have rated a total of 83,509 items. The total number of ratings is 631,064, and the total number of the issued trust statements is 511,799. Note that the user-item rating matrix of Epinions dataset is quite sparse, and its density is around 0.015%. The statistics of user-item matrix and user-trust matrix are summarized into Table 1 and Table 2;

The different amounts of training data (80% and 90%) are used to evaluate the algorithms. The training data 80% means that we randomly choose 80% of ratings from the Epinions dataset as the training data to predict the ratings of the remaining 20%. The random selection was performed five times independently. For the proposed method, the regularization coefficient λ is empirically set to one, and η is 0.1. All of our experiments were carried on a PC with Intel 2.8GHz processor and 4GB RAM.

5.2 Performance Measure

Both the Root Mean Square Error (RMSE) and the Mean Absolute Error (MAE) are employed as the performance measure to evaluate the prediction quality of our proposed approaches in comparison with other related recommendation methods. More specifically, the metrics RMSE is defined as follows:

$$\mathbf{RMSE} = \sqrt{\frac{1}{p}\sum_{i,j}(M_{i,j} - \hat{M}_{i,j})^2}$$

where $M_{i,j}$ denotes the rating on the item j annotated by the user i, and $\hat{M}_{i,j}$ is the predicted rating for the user i on the item j. p is denoted as the total number of ratings in the testing dataset. Additionally, the metrics MAE is defined as below:

$$\mathbf{MAE} = \frac{1}{p}\sum_{i,j}|M_{i,j} - \hat{M}_{i,j}|$$

To make a consistent comparison with the previous methods, we employ the inverse mapping function of $g(x)$ in Eqn. 2 to transform the predicted rating value in our method back into the original range.

5.3 Experiments on Epinions Dataset

The rank d is an important parameter for the matrix factorization problem. Instead of assigning it to an empirical value like the previous methods, we can estimate d based on our LRSDP formulation. For the proposed approach, we observe that the number of variables is $d(m+n)$ and the total number of equality constraints is equal to the number of ratings p. To guarantee a well-posed problem, the rank d should be less than $\frac{p}{m+n}$. Moreover, we did not take into account of both the inactive users and those items having no ratings in practice. Therefore, the rank d is set to five.

In this paper, we employ the user-trust information from Epinions dataset to build the graph Laplacian matrix L. As discussed in Section 4.1, we have three choices on selecting the kernel function for the graph Laplacian. Table 3 presents the experimental results of different settings. From the empirical results, we first observe that both the distance kernel and heat kernel outperform the binary kernel, which indicates that it is effective to incorporate the users' taste by comparing their ratings. Second, the distance kernel function using the modified cosine similarity achieves the best performance in the empirical evaluation. Therefore, we choose the modified cosine distance function to build the graph Laplacian matrix L in the following experiments.

Table 3: Evaluation on the different kernel functions (A Smaller RMSE or MAE Value Means a Better Performance)

Training Data	Metrics	Binary	Distance	Heat
80%	RMSE	1.1379	**1.1304**	1.1323
	MAE	0.8639	**0.8557**	0.8606
90%	RMSE	1.1155	**1.1095**	1.1197
	MAE	0.8540	**0.8338**	0.8457

Then, we conduct the empirical comparisons to investigate the effectiveness of the logistic function $\rho(x)$ for normalization. Moreover, we study the LRSDP method without the graph Laplacian regularization, which is equivalent to the low-rank semidefinite program solution for the minimization problem in Eqn. 3. To make it clear, the proposed LRSDP with the graph Laplacian regularization approach is denoted as "LRSDP". The LRSDP method without normalization is denoted as "LRSDP(w/o ρ)". Similarly, the LRSDP method without the graph Laplacian regularization is denoted as "LRSDP(w/o L)". Table 4 shows the experimental results for the proposed methods and the state-of-the-art approaches. We can observe that LRSDP(w/o ρ) obtains the very poor results, which reveals the significance of normalization using the logistic function. Moreover, the proposed LRSDP approach outperforms the two recent social recommendation approaches: STE and SoRec. This demonstrates that the graph Laplacian regularization not only stands on a solid theoretical framework but also obtains the promising results in practice. Looking into the performance comparisons, we can also find that LRSDP(w/o L) performs slightly better than its counterpart PMF method using gradient descent optimization.

Finally, we empirically study the efficiency performance of the proposed LRSDP method. Table 5 summarize the computational time for factorizing the user-item rating matrix using STE and LRSDP. From these results, it can be clearly observed that the proposed LRSDP approach is much

Table 4: Performance Comparisons (A Smaller RMSE or MAE Value Means a Better Performance)

Training Data	Metrics	PMF	SoRec	Trust	STE	LRSDP (w/o ρ)	LRSDP (w/o L)	LRSDP
80%	RMSE	1.1826	1.1530	1.2140	1.1346	1.4998	1.1502	**1.1304**
	MAE	0.8951	0.8638	0.9221	0.8594	1.1730	0.8830	**0.8557**
90%	RMSE	1.1575	1.1333	1.1959	1.1109	1.5677	1.1292	**1.1095**
	MAE	0.8676	0.8442	0.9054	0.8377	1.2219	0.8593	**0.8338**

more efficient than the STE method (Ma, King, and Lyu 2009). Specifically, LRSDP only requires no more than 7% of the training time for STE. This is because the LRSDP approach employs an efficient quasi-Newton optimization algorithm while the STE method involves with the time-consuming step to directly fuse the social trust information into the high dimensional user-item matrix. Additionally, we can see that the computational time for the presented LRSDP method increases along with the total number of ratings in the user-item matrix.

Table 5: Comparisons of time cost on Epinions dataset

STE (90%)	LRSDP (80%)	LRSDP (90%)
133min	7.5min	8.3min

6 Conclusions

It is clear that our novel low-rank semidefinite program approach to social recommendation is powerful and effective. It offers several distinct advantages over the conventional approaches. First, we introduce the graph Laplacian to effectively regularize the user-specific latent space and capture the underlying relationships among the different users. Second, the presented social recommendation with the graph Laplacian regularization problem is directly formulated into the low-rank semidefinite programming, which can be efficiently solved by the quasi-Newton algorithm. Finally, the mapping function for the normalization is carefully addressed in our formulation. Our approach has been tested on the Epinions dataset with over half million ratings. The encouraging experimental results show that our presented method is both effective and promising.

In the future, we will investigate the relationship among the items by taking into account of the category information. Moreover, we will explore the recommendation problem in the multimedia domain, in which the content information in music and videos can be used to estimate the similarity between the different items.

Acknowledgments

The work was fully supported by Program for New Century Excellent Talents in University (NCET-09-0685) and Fundamental Research Funds for the Central Universities.

References

Boyd, S., and Vandenberghe, L. 2004. *Convex Optimization*. Cambridge University Press.

Breese, J. S.; Heckerman, D.; and Kadie, C. 1998. Empirical analysis of predictive algorithms for collaborative filtering. In *Proc. of UAI '98*.

Burer, S., and Choi, C. 2006. Computational enhancements in low-rank semidefinite programming. *Optimization Methods and Software* 21:493–512.

Burer, S., and Monteiro, R. D. 2003. A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization. *Mathematical Programming* 95:329–357.

Cvetkovic, D. M.; Doob, M.; and Sachs, H. 1998. *Spectra of graphs : theory and application*. New York: Wiley.

Hofmann, T. 2003. Collaborative filtering via gaussian probabilistic latent semantic analysis. In *Proc. of SIGIR '03*, 259–266. New York, NY, USA: ACM.

Jamali, M., and Ester, M. 2010. A matrix factorization technique with trust propagation for recommendation in social networks. In *Proceedings of the fourth ACM conference on Recommender systems*, RecSys '10, 135–142.

Jin, R.; Chai, J. Y.; and Si, L. 2004. An automatic weighting scheme for collaborative filtering. In *Proc. of SIGIR '04*, 337–344. New York, NY, USA: ACM.

Li, W.-J., and Yeung, D.-Y. 2009. Relation regularized matrix factorization. In *Proceedings of the 21st international jont conference on Artifical intelligence*, 1126–1131.

Linden, G.; Smith, B.; and York, J. 2003. Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet Computing* 76–80.

Ma, H.; Yang, H.; Lyu, M. R.; and King, I. 2008. Sorec: social recommendation using probabilistic matrix factorization. In *Proceeding of the 17th ACM conference on Information and knowledge management*, CIKM '08, 931–940.

Ma, H.; King, I.; and Lyu, M. R. 2009. Learning to recommend with social trust ensemble. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '09, 203–210.

Mitra, K.; Sheorey, S.; and Chellappa, R. 2010. Large-scale matrix factorization with missing data under additional constraints. In *Advances in Neural Information Processing Systems 23*. 1642–1650.

Salakhutdinov, R., and Mnih, A. 2008. Probabilistic matrix factorization. In *Advances in Neural Information Processing Systems*, volume 20.

Si, L., and Jin, R. 2003. Flexible mixture model for collaborative filtering. In *Proc. of ICML '03*.

Srebro, N.; Rennie, J. D. M.; and Jaakkola, T. S. 2005. Maximum-margin matrix factorization. *Advances in Neural Information Processing Systems* 17:1329–1336.

Zhang, Y., and Koren, J. 2007. Efficient bayesian hierarchical user modeling for recommendation system. In *Proc. of SIGIR '07*, 47–54. New York, NY, USA: ACM.