The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

Social Recommendation with an Essential Preference Space

Chun-Yi Liu,[1,2] Chuan Zhou,[1,2] Jia Wu,[3] Yue Hu,[1,2] Li Guo[1,2]
[1]Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
[2]School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
[3]Department of Computing, Macquarie University, Sydney, NSW 2109, Australia
{liuchunyi, zhouchuan, huyue, guoli}@iie.ac.cn, jia.wu@mq.edu.au

Abstract

Social recommendation, which aims to exploit social information to improve the quality of a recommender system, has attracted an increasing amount of attention in recent years. A large portion of existing social recommendation models are based on the tractable assumption that users consider the same factors to make decisions in both recommender systems and social networks. However, this assumption is not in concert with real-world situations, since users usually show different preferences in different scenarios. In this paper, we investigate how to exploit the differences between user preference in recommender systems and that in social networks, with the aim to further improve the social recommendation. In particular, we assume that the user preferences in different scenarios are results of different linear combinations from a more underlying user preference space. Based on this assumption, we propose a novel social recommendation framework, called social recommendation with an essential preferences space (SREPS), which simultaneously models the structural information in the social network, the rating and the consumption information in the recommender system under the capture of essential preference space. Experimental results on four real-world datasets demonstrate the superiority of the proposed SREPS model compared with seven state-of-the-art social recommendation methods.

Introduction

Social recommendation, which aims to incorporate social relations into recommender systems, has attracted more and more attention in recent years. Previous studies have demonstrated the potential of social relations to improve recommendation performance and alleviate sparsity and cold-start problems in recommender systems (Guo, Zhang, and Thalmann 2012; Guo, Zhang, and Yorke-Smith 2015; Gao et al. 2017). These studies are mainly based on the assumption that user preferences are similar to his/her neighbors.

A large proportion of previous studies (Ma et al. 2008; Tang et al. 2013; Yang et al. 2013; Rafailidis and Crestani 2016) collectively factorize the rating matrix and the social relationship matrix, sharing the same latent vectors to characterize the user preferences in both item rating and social relationship. However, this method is not always reasonable in real-world situations, since users may show different preferences in different scenarios. A more intelligent way is to involve bias vectors between the user latent vectors in historical ratings and social relationships (Hsieh et al. 2016). Although this method incurs different preferences, the two kinds of latent vectors still belong to the same latent space. Naturally, users in different real-world scenarios should consider completely different latent factors, with an explanatory example shown in Fig. 1. In summary, the user latent vectors in recommender system and social network should belong to different latent spaces rather than two different vectors in the same latent space.

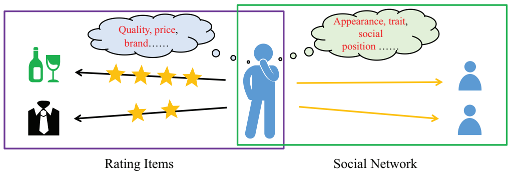


Figure 1: An explanatory example. The purple box is the scenarios that the user rates items. The user mainly considers quality, price and brand to make decisions. The green box is the scenarios in social networks, and the user regards appearance, personality traits and social position to decide whether to trust other people. Clearly, the user considers different latent factors in these two scenarios.

To address this issue, in this paper we introduce an **essential preference space** to describe the user multiple preferences. The user latent spaces in recommender system and the social network are assumed as different projections from the essential preference space. Based on this assumption, we propose a novel framework called SREPS, where the rating information, the consumption information (i.e., whether a user rated an item) in the recommender system and the structure information in the social network are utilized to build user differential latent vectors for different scenarios from the essential preference space. Specifically, *rating information* is modeled by well-studied matrix factorization. A state-of-the-art network embedding model called large-scale information network embedding (LINE) (Tang et al. 2015) is adopted to exploit the sparse *structural information* of the social network. *Consumption information* is viewed as a bipartite network. To model the three kinds of information jointly, a number of space projection matrices are involved to map the essential preference space into the different latent spaces learned from the rating, consumption and social network structure information. A stochastic gradient descent algorithm is adopted for the SREPS model learning.

Our contributions are summarized below:

- A novel essential preference space is introduced to describe the user multiple preference differences in different scenarios like recommender system and social network.
- A social recommendation framework SREPS is proposed to jointly model the rating, consumption and social relation information based on the essential preference space. An effective stochastic gradient descent algorithm is designed for the model learning.
- Experimental results on four real-world datasets demonstrate the effectiveness of the proposed SREPS model compared with seven state-of-the-art baselines.

Preliminaries and Problem Definitions

Problem Definition

The recommendation problem in this paper is to predict the rating that a user will provide to an item that he has not rated based on his historical ratings and his social network.

Assume that a recommender system includes a user set P with m users and an item set Q with n items. Let $R = [r_{ui}]_{m\times n}$ denote the user-item rating matrix, where entry r_{ui} is the rating that user u rated item i. Note that in the rating matrix R, only a small portion of the ratings are observed and the indexes constitute the observed index set $\Omega = \{(u,i)|(u,i) \text{ is observed.}\}$. The other entities are unknown. We aim to predict the values of the unknown entities, i.e., the ratings that users will give to items they have not yet rated.

Social recommendation can't do without social network. Suppose that a social network is represented by a graph $\mathcal{G}^{\mathcal{S}} = \left(\mathcal{V}^{\mathcal{S}}_{\mathcal{P}}, \mathcal{E}^{\mathcal{S}}\right)$, where $\mathcal{V}^{\mathcal{S}}_{\mathcal{P}}$ is the set of vertexes that represent users, and $\mathcal{E}^{\mathcal{S}}$ is the set of edges, which can be both directed (e.g., trust) and undirected (e.g., friendship). We use network embedding to preserve the network properties and obtain a low-dimensional latent representation of each vertex in the social network.

We also establish a recommendation network from the historical rating information. A recommendation network is a bipartite graph $\mathcal{G}^{\mathcal{R}} = \left(\mathcal{V}^{\mathcal{R}}_{\mathcal{P}}, \mathcal{V}^{\mathcal{R}}_{\mathcal{Q}}, \mathcal{E}^{\mathcal{R}}\right)$, where $\mathcal{V}^{\mathcal{R}}_{\mathcal{P}}$ is a set of vertexes to represent users, $\mathcal{V}^{\mathcal{R}}_{\mathcal{Q}}$ is another set of vertexes to represent items, and $\mathcal{E}^{\mathcal{R}}$ is the set of *directed unweighted* edges from users to items. The directed edge $e = (u,i) \in \mathcal{E}^{\mathcal{R}}$ denotes that user u has rated item i.

As claimed in (Ohsawa, Obara, and Osogami 2016), the rating actions (i.e., whether to rate the item) can implicitly express user preferences in the sense that users with similar rated items may probably have similar implicit preferences. Along this idea, we explore the structure information of the recommendation network to learn the user implicit preferences with the help of property-reserved network embedding method.

In sum, our goal is to use the historical ratings in matrix R, social network $\mathcal{G}^{\mathcal{S}}$, and recommendation network $\mathcal{G}^{\mathcal{R}}$ to predict the unknown entities in matrix R.

Matrix Factorization

A matrix factorization based recommender method is used as our basic model. Matrix factorization based methods are widely used in social recommendation models. Let U_u and V_i be the d_0-dimensional latent (column) vectors for user u and item i respectively. Matrix factorization aims to find the low-dimensional latent vectors with which to model the user preferences. The unknown entities then be predicted by calculating the inner products of these latent vectors, i.e., $r_{ui} = U_u^T V_i$, where U_u^T is the transpose of latent vector U_u. Formally, the loss function for matrix factorization is as follows,

$$\min_{U_u,V_i} \frac{1}{2}\sum_{(u,i)\in\Omega}\left(r_{ui} - U_u^T V_i\right)^2 + \frac{\lambda}{2}\left(\sum_u \|U_u\|^2 + \sum_i \|V_i\|^2\right) \tag{1}$$

where the second term controlled by λ is to avoid overfitting.

Large-scale Information Network Embedding

LINE (Tang et al. 2015) is a state-of-the-art network embedding model and shows better performance than graph matrix factorization when exploring network structural information. LINE captures both the local structures and similarity of neighborhood network structures between two vertexes. Noting that each edge in an undirected network can be viewed as two directed edges with opposite directions, we assume that the considered network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is a directed network. In LINE, each vertex can be treated as a context for the other vertexes, and vertexes with similar distributions over a context are assumed to be similar. For each vertex $s \in \mathcal{V}$, there exists an embedding vector $E_s \in \mathbb{R}^{d_1}$ when s plays the role of a vertex itself, and a context vector $C_s \in \mathbb{R}^{d_1}$ when s is a context of other vertexes. For each directed edge (s,t) with weight w_{st}, the probability of context t generated from vertex s is defined as

$$p(t|s) = \frac{\exp\left(C_t^T E_s\right)}{\sum_{v\in\mathcal{V}}\exp\left(C_v^T E_s\right)} \tag{2}$$

The empirical probability is that $\hat{p}(t|s) = \frac{w_{st}}{d_s^{out}}$, where d_s^{out} is the out-degree of vertex s, i.e., $d_s^{out} = \sum_{v\in\mathcal{V}} w_{sv}$. The objective function of LINE is defined as

$$\min_{E_s,C_s}\sum_{s\in\mathcal{V}} d_s^{out}\mathrm{KL}\left(\hat{p}(\cdot|s), p(\cdot|s)\right)$$

where KL(p,q) is the KL-divergence of the probability distributions p and q, $\hat{p}(\cdot|s)$ and $p(\cdot|s)$ are the empirical and defined distributions of contexts generated from vertex s, respectively. Omitting the constants, which does not affect the optimization of the objective function, the final loss function can be put as

$$\min_{E_s,C_s} -\sum_{(s,t)\in\mathcal{E}} w_{st}\log p(t|s) \tag{3}$$

By incorporating Eq. (2), we can minimize Eq. (3) to obtain the optimal embedding and context vectors.

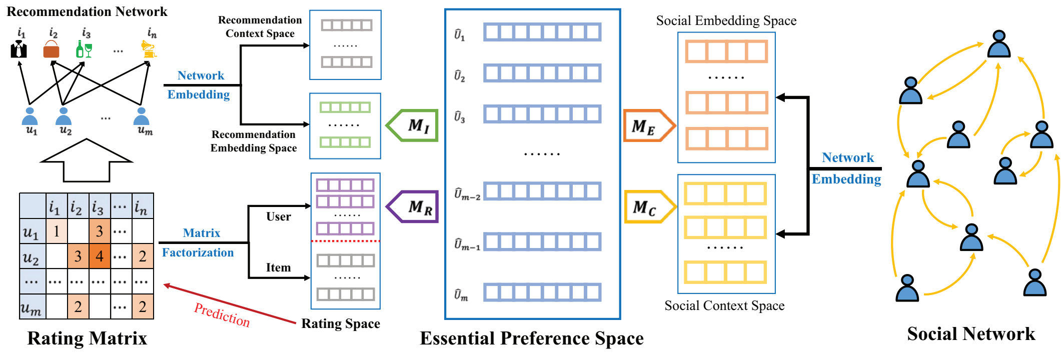


Figure 2: The overview of our SREPS model. Each user has a latent vector in the essential preference space, and his semantic latent vectors are projections from the essential preference space by multiplying space projection matrices (i.e. M_E, M_C, M_R and M_I). We model the historical rating information with matrix factorization, while the social and recommendation networks are modeled by network embedding. By jointly modeling these elements, we can learn the user latent vectors in the essential preference space and the space projection matrices. Finally, we can use user latent vectors in essential preference space, the rating space projection matrix M_R, and the item latent vectors in the rating space to predict the final rating.

Social Recommendation with an Essential Preference Space (SREPS)

Essential Preference Space

The definitions for the semantic latent space and the essential preference space are first present below.

Definition 1 ***Semantic Latent Space.*** *For a particular scenario like item rating and friend trusting, the corresponding semantic latent space is inferred from the user feedback and can be used to explain the user preferences by characterizing users in terms of latent factors. For example, the user latent vectors learned from Eq.* (1) *belong to a rating semantic latent space, and the embedding learned representation vectors from Eq.* (3) *belong to a social semantic latent space.*

Definition 2 ***Essential Preference Space.*** *The essential preference space is used to describe the fundamental factors that influence user preferences. The factor in each semantic latent space is a linear combination of factors in the essential preference space. The transformation from factors in essential preference space to factors in semantic latent space can be operated by multiplying space projection matrices.*

Let $\hat{U}_u \in \mathbb{R}^l$ be the latent vector in essential preference space for user u. Let $U_u \in \mathbb{R}^{d_0}$ be the latent vector in rating semantic latent space for user u in Eq. (1), which can be obtained from following transition

$$U_u = M_R \hat{U}_u, \tag{4}$$

where $M_R \in \mathbb{R}^{d_0 \times l}$ is the space projection matrix that maps the essential preference space into the rating semantic latent space.

Similarly, the embedding vector E_u and context vector C_u in Eq. (3) can be mapped from $\hat{U}_u$ by space projection matrices $M_E \in \mathbb{R}^{d_1 \times l}$ and $M_C \in \mathbb{R}^{d_1 \times l}$ as follow:

$$E_u = M_E \hat{U}_u, \; C_u = M_C \hat{U}_u. \tag{5}$$

The traditional social recommendation models, which shared the common user latent vector in both recommender systems and social networks, are special cases of our essential preference space model, when the dimensions of essential preference space and semantic latent spaces are identical and all space projection matrices are identity ones.

The SREPS Model

With the notations of above essential preference space and space projection matrices, the SREPS model is formulated as follows.

By incorporating Eq. (4) and Eq. (5), the rating loss function in Eq. (1) without regularizations can be represented as

$$O_1 = \frac{1}{2} \sum_{(u,i) \in \Omega} \left(r_{ui} - \hat{U}_u^T M_R^T V_i \right)^2 \tag{6}$$

The loss function O_2 for the social network representation is as follows

$$O_2 = - \sum_{(s,t) \in \mathcal{E}^{\mathcal{S}}} w_{st} \log \frac{\exp\left(\hat{U}_t^T M_C^T M_E \hat{U}_s \right)}{\sum_{v \in \mathcal{V}_{\mathcal{P}}^{\mathcal{S}}} \exp\left(\hat{U}_v^T M_C^T M_E \hat{U}_s \right)} \tag{7}$$

where w_{st} is the weight in edge (s, t). Similarly, the loss function O_3 for the recommendation network representation is expressed as

$$O_3 = - \sum_{(u,i) \in \Omega} \log \frac{\exp\left(B_i^T M_I \hat{U}_u \right)}{\sum_{v \in \mathcal{V}_{\mathcal{Q}}^{\mathcal{R}}} \exp\left(B_v^T M_I \hat{U}_u \right)} \tag{8}$$

where $M_I \in \mathbb{R}^{d_2 \times l}$ is the space projection matrix corresponding to recommendation network and $B_i \in \mathbb{R}^{d_2}$ is the context vector of item i. Note that the recommendation network is a bipartite graph. Hence, different from the social network, the user vertexes have only the embedding vectors, while the item vertexes have only context vectors.

In sum, the loss function of the SREPS model is

$$L = (1 - \alpha - \beta) O_1 + \alpha O_2 + \beta O_3 + Reg \tag{9}$$

where $\alpha \geqslant 0$ and $\beta \geqslant 0$ are parameters that control the balance of loss function meeting $\alpha + \beta \leqslant 1$, and Reg is the regularization term:

$$Reg = \frac{\lambda}{2}\left(\sum_i \left(\|V_i\|^2 + \|B_i\|^2\right) + \sum_u \left(\|M_R\hat{U}_u\|^2 \right.\right. \\ \left.\left. + \|M_E\hat{U}_u\|^2 + \|M_C\hat{U}_u\|^2 + \|M_I\hat{U}_u\|^2 + \|\hat{U}_u\|^2\right)\right) \tag{10}$$

where λ is the regularization parameter. To reduce the model complexity, the same regularization parameter λ is used for all the variables. One may find that the space projection matrices M_C and M_E appear only in the product $M_C^T M_E$, which can thus be substituted into a new matrix to reduce the parameter number. However, we here adopt the intuitive product form in Eq. (7) to better show our basic ideas of essential preference space.

Prediction

In the SREPS model, the final rating r_{ui} that user u provides to an unrated item i can be predicted from the preference in the rating semantic latent space by $\hat{U}_u$, M_R and V_i as follows:

$$r_{ui} = \hat{U}_u^T M_R^T V_i \tag{11}$$

Note that the predicted ratings may fall out of the rating range. To avoid this situation, we adopt the following method to project the predicted ratings into the rating range: if $r_{ui} > r_{max}$, $r_{ui} = r_{max}$, and if $r_{ui} < r_{min}$, $r_{ui} = r_{min}$, where r_{max} and r_{min} are the upper and lower bounds of the rating range.

Optimization Approach

The loss function in Eq. (9) is the combination of three loss functions. Inspired by (Krohn-Grimberghe et al. 2012), we simultaneously learn the parameters by sampling examples from different parts of the SREPS loss function.

Rating Loss We randomly sample a pair (u, i) from the observed entity set Ω. We only consider the regularizations that directly affect the rating loss function in Eq. (6), i.e., $\|V_i\|^2$, $\|M_R\hat{U}_u\|^2$, and $\|\hat{U}_u\|^2$. Thus, the gradients of the rating loss function $L_1 := O_1 + Reg_1$ for the sampled pair (u, i) are as follow,

$$\frac{\partial L_1}{\partial \hat{U}_u} = (1-\alpha-\beta)\delta_{ui}^R M_R^T V_i + \lambda\left(I_l + M_R^T M_R\right)\hat{U}_u \tag{12}$$

$$\frac{\partial L_1}{\partial V_i} = (1-\alpha-\beta)\delta_{ui}^R M_R \hat{U}_u + \lambda V_i \tag{13}$$

$$\frac{\partial L_1}{\partial M_R} = (1-\alpha-\beta)\delta_{ui}^R V_i \hat{U}_u^T + \lambda M_R \hat{U}_u \hat{U}_u^T \tag{14}$$

where I_l is an $l \times l$ identity matrix and $\delta_{ui}^R = \hat{U}_u^T M_R^T V_i - r_{ui}$.

Social Network Embedding Now we optimize the loss function $L_2 := O_2 + Reg_2$ in the social network embedding, where Reg_2 is the regularization corresponding to $\|M_E\hat{U}_u\|^2$, $\|M_C\hat{U}_u\|^2$, and $\|\hat{U}_u\|^2$. Since the $p(t|s)$ in Eq. (2) requires the summation of the entire set of vertexes, optimizing O_2 is computationally expensive, even if we sample an edge (t, s) from the edge set $\mathcal{E}^{\mathcal{S}}$. To tackle this problem, we adopt the negative sampling method (Mikolov et al. 2013) which is used to distinguish the target vertex from negative vertexes generated from the noise distribution, i.e., we can change $p(t|s)$ into the following form,

$$p(t|s) \propto \log\sigma\left(C_t^T E_s\right) + \sum_{i=1}^{K} E_{n_i}\left[\log\sigma\left(-C_{n_i}^T E_s\right)\right] \tag{15}$$

where $\sigma(\cdot)$ is the sigmoid function, K is the number of negative samples, and the negative vertexes v_{n_i} are drawn from the distribution $P_n(v)$. Here we set $P_n(v) \propto d_v^{3/4}$, where d_v is the out-degree of vertex v. Empirically, we set $K = 5$. Thus, for the randomly sampled edge (t, s), we can obtain the gradients as follow,

$$\hat{U}_{stn} = \delta_{st}^S \hat{U}_t + \sum_{i=1}^{K} \delta_{sn_i}^S \hat{U}_{n_i} \tag{16}$$

$$\frac{\partial L_2}{\partial \hat{U}_s} = \alpha M_E^T M_C \hat{U}_{stn}^T + \lambda\left(I_l + M_E^T M_E\right)\hat{U}_s \tag{17}$$

$$\frac{\partial L_2}{\partial \hat{U}_t} = \alpha\delta_{st}^S M_C^T M_E \hat{U}_s + \lambda\left(I_l + M_C^T M_C\right)\hat{U}_t \tag{18}$$

$$\frac{\partial L_2}{\partial \hat{U}_{n_i}} = \alpha\delta_{sn_i}^S M_C^T M_E \hat{U}_s + \lambda\left(I_l + M_C^T M_C\right)\hat{U}_{n_i} \tag{19}$$

$$\frac{\partial L_2}{\partial M_E} = \alpha M_C \hat{U}_{stn} \hat{U}_s^T + \lambda M_E \hat{U}_s \hat{U}_s^T \tag{20}$$

$$\frac{\partial L_2}{\partial M_C} = \alpha M_E \hat{U}_s \hat{U}_{stn}^T + \lambda M_C\left(\hat{U}_t\hat{U}_t^T + \sum_{i=1}^{K}\hat{U}_{n_i}\hat{U}_{n_i}^T\right) \tag{21}$$

where $\delta_{st}^S = \sigma\left(\hat{U}_t^T M_C^T M_E \hat{U}_s\right) - 1$ and $\delta_{sn_i}^S = \sigma\left(\hat{U}_{n_i}^T M_C^T M_E \hat{U}_s\right)$.

Recommendation Network Embedding The remaining expression in Eq. (9) is the recommendation network embedding objective $L_3 := O_3 + Reg_3$, where Reg_3 contains $\|B_i\|^2$, $\|M_I\hat{U}_u\|^2$, and $\|\hat{U}_u\|^2$. The negative sampling method is also adopted in the learning process by replacing C_t, C_{n_i} and E_s with B_i, B_{n_j} and $M_I\hat{U}_u$ respectively. Since the recommendation network is a bipartite network, and we simply sample the negative item vertexes according to a uniform distribution. Similarly, we can obtain the gradients for each edge $(u, i) \in \mathcal{E}^{\mathcal{R}}$ as follow

$$\frac{\partial L_3}{\partial \hat{U}_u} = \beta M_I^T\left(\delta_{ui}^I B_i + \sum_{j=1}^{K}\delta_{un_j}^I B_{n_j}\right) + \lambda\left(I_l + M_I^T M_I\right)\hat{U}_u \tag{22}$$

$$\frac{\partial L_3}{\partial B_i} = \beta\delta_{ui}^I M_I \hat{U}_u + \lambda B_i \tag{23}$$

$$\frac{\partial L_3}{\partial B_{n_j}} = \beta\delta_{un_j}^I M_I \hat{U}_u + \lambda B_{n_j}, \forall i\ 1 \leqslant i \leqslant K, \tag{24}$$

$$\frac{\partial L_3}{\partial M_I} = \beta\left(\delta_{ui}^I B_i + \sum_{j=1}^{K}\delta_{un_j}^I B_{n_j}\right)\hat{U}_u^T + \lambda M_I \hat{U}_u \hat{U}_u^T \tag{25}$$

where $\delta_{ui}^I = \sigma\left(B_i^T M_I \hat{U}_u\right) - 1$ and $\delta_{un_j}^I = \sigma\left(B_{n_j}^T M_I \hat{U}_u\right)$.

Experiments

We conduct experiments on four real-world data sets to evaluate the performance of the proposed SREPS model.

Experimental Settings

Dataset Four datasets were used in our experiments: FilmTrust (Guo, Zhang, and Yorke-Smith 2013), Flixster (Jamali and Ester 2010), Epinions (Tang, Gao, and Liu 2012) and Ciao (Tang et al. 2012). These datasets contain both item ratings and social relationships. Flixster is undirected and can be regarded as a directed graph by treating an undirected edge as bidirectional. A subset of Flixster was randomly sampled to be used as the dataset in this paper. The dataset statistics are presented in Table 1.

Table 1: Statistics of the datasets

Feature	FilmTrust	Flixster	Epinions	Ciao
#User	1,508	53,004	22,164	7,375
#Item	2,071	18,144	296,277	105,114
#Rating	35,497	409,243	922,267	284,086
#Social Link	1,853	613,509	355,754	111,781

For each dataset, 80% of the rating data are selected randomly as the training set and the rest are used as the testing set. We repeated each experiment 10 times and report the average performance and standard deviation.

Evaluation Metrics We adopted two representative metrics to evaluate the performance: mean absolute error (MAE) and root mean square error (RMSE). A smaller MAE or RMSE value means better performance. Even a small improvement in MAE and RMSE values can have a significant impact on the quality of the top-few recommendations (Koren 2008).

Comparison Methods

We evaluated the effectiveness of the proposed SREPS model by comparing it with the following seven state-of-the-art social recommendation models:

- **PMF** (Salakhutdinov and Mnih 2007) only uses rating information to factorize the user-item rating matrix under the probabilistic framework.
- **SoRec** (Ma et al. 2008) jointly factorizes the user-item rating matrix and the user-user social relation matrix, and shares the same user latent factors.
- **STE** (Ma, King, and Lyu 2009) models user ratings as a combination of a user's preferences and his social neighbors within the matrix factorization framework.
- **SocialMF** (Jamali and Ester 2010) adds social regularization that regularizes the user latent vector to be similar to the average of those of his social neighbors.
- **SoReg** (Ma et al. 2011) minimizes the sum of the weighted differences between user latent vectors as social regularization.
- **TrustMF** (Yang et al. 2013) jointly factorizes the user-item rating matrix and user-user social matrix from truster and trustee perspectives.
- **SoDimRec** (Tang et al. 2016) considers the heterogeneity and weak dependency connections in the social network, and models the two aspects as social regularization terms.

Table 2: Parameter Settings of the Comparison Models

Models	Parameters	FilmTrust	Flixster	Epinions	Ciao
SoRec	λ_C	0.1	0.01	0.3	0.01
STE	α	1	1	0.4	1
SocialMF	λ_T	1	1	1	1
SoReg	β	0.3	1	0.1	0.1
TrustMF	λ_T	1	1	1	1
SoDimRec	c	50	500	500	100
	λ_1	5	10	10	5
	λ_2	50	100	100	100
SREPS	α	0.2	0.3	0.4	0.3
	β	0.1	0.1	0.1	0.2

The optimal experimental settings for each method were either determined by our experiments or were taken from the suggestions by previous works. The setting that were taken from previous works include: the learning rate $\eta = 0.001$; and the dimension of the latent vectors $d = 5$ and 10. All the regularization parameters for the latent vectors were set to be the same at $\lambda_U = \lambda_V = 0.001$. The other parameters are shown in Table 2.

We set $l = d_0 = d_1 = d_2$ for the SREPS model, i.e., the dimensions of the essential preference space and the three semantic latent spaces were the same. The regularization parameter λ was set to be 0.001. The hyper parameters α and β are also shown in Table 2 and were based on the results of the parameter sensitivity analyses.

Results and Analysis

The comparison results are provided in Table 3 with the following observations:

- The models with social networks outperform rating-based PMF, which demonstrates that exploiting social network information can improve the performance of recommender systems evaluated by both MAE and RMSE.
- The proposed SREPS outperforms SoRec and TrustMF. These two models are mainly based on collective matrix factorization by sharing the user latent factors. These models can be specific situations in the essential preference space model, i.e., our SREPS is able to model more general situations than these two models. Moreover, the network embedding methods are able to not only capture the similarity of users with direct social links, but also those with similar social structures, i.e., the network embedding method can obtain more information than matrix factorization approaches from social networks.
- Compared to STE, SocialMF, and SoReg, SREPS achieved the best performance. STE models ratings as a combination of a user's preference and the weighted average of his social neighbors. SocialMF and SoReg add social regularization terms into the matrix factorization from different perspectives. The common consideration of these three models is that user preferences in a recommender system should be very similar to those of their social neighbors. SREPS can model similar users while preserving the differences between user actions in differ-

Table 3: Experimental Results and Standard Deviation (the Best Scores are in Bold)

Datasets	Metrics	PMF	SoRec	STE	SocialMF	SoReg	TrustMF	SoDimRec	SREPS
FilmTrust d=5	MAE	0.688 (± 0.0087)	0.651 (± 0.0092)	0.648 (± 0.0080)	0.641 (± 0.0013)	0.675 (± 0.0038)	0.641 (± 0.0011)	0.639 (± 0.0052)	**0.627** **(± 0.0017)**
	RMSE	0.956 (± 0.0080)	0.913 (± 0.0090)	0.905 (± 0.0079)	0.884 (± 0.0027)	0.936 (± 0.0123)	0.885 (± 0.0014)	0.884 (± 0.0107)	**0.866** **(± 0.0031)**
Flixster d=5	MAE	0.816 (± 0.0118)	0.754 (± 0.0087)	0.753 (± 0.0088)	0.777 (± 0.0019)	0.825 (± 0.0049)	0.898 (± 0.0015)	0.798 (± 0.0060)	**0.726** **(± 0.0020)**
	RMSE	1.077 (± 0.0090)	0.981 (± 0.0085)	0.983 (± 0.0080)	0.999 (± 0.0028)	1.094 (± 0.0149)	1.151 (± 0.0022)	1.064 (± 0.0144)	**0.944** **(± 0.0033)**
Epinions d=5	MAE	0.984 (± 0.0168)	0.891 (± 0.0135)	0.958 (± 0.0113)	0.832 (± 0.0020)	0.949 (± 0.0053)	0.819 (± 0.0014)	0.811 (± 0.0063)	**0.809** **(± 0.0023)**
	RMSE	1.302 (± 0.0104)	1.123 (± 0.0112)	1.197 (± 0.0112)	1.077 (± 0.0027)	1.220 (± 0.0132)	1.075 (± 0.0019)	1.067 (± 0.0124)	**1.041** **(± 0.0002)**
Ciao d=5	MAE	0.926 (± 0.0105)	0.773 (± 0.0093)	0.771 (± 0.0100)	0.757 (± 0.0016)	0.907 (± 0.0051)	0.757 (± 0.0010)	0.745 (± 0.0056)	**0.728** **(± 0.0016)**
	RMSE	1.216 (± 0.0112)	1.021 (± 0.0114)	1.029 (± 0.0082)	0.990 (± 0.0029)	1.190 (± 0.0147)	0.990 (± 0.0020)	0.977 (± 0.0102)	**0.960** **(± 0.0033)**
FilmTrust d=10	MAE	0.677 (± 0.0101)	0.638 (± 0.0076)	0.643 (± 0.0071)	0.625 (± 0.0013)	0.668 (± 0.0041)	0.638 (± 0.0010)	0.625 (± 0.0044)	**0.615** **(± 0.0018)**
	RMSE	0.917 (± 0.0088)	0.886 (± 0.0090)	0.891 (± 0.0076)	0.869 (± 0.0021)	0.902 (± 0.0108)	0.879 (± 0.0018)	0.868 (± 0.0120)	**0.845** **(± 0.0025)**
Flixster d=10	MAE	0.771 (± 0.0109)	0.791 (± 0.0094)	0.788 (± 0.0094)	0.786 (± 0.0017)	0.789 (± 0.0055)	0.826 (± 0.0015)	0.782 (± 0.0058)	**0.727** **(± 0.0016)**
	RMSE	1.019 (± 0.0084)	1.025 (± 0.0088)	1.023 (± 0.0097)	1.017 (± 0.0027)	1.041 (± 0.0150)	0.959 (± 0.0030)	1.007 (± 0.0122)	**0.951** **(± 0.0031)**
Epinions d=10	MAE	0.914 (± 0.0144)	0.887 (± 0.0106)	0.967 (± 0.0116)	0.833 (± 0.0016)	0.940 (± 0.0063)	0.810 (± 0.0044)	0.823 (± 0.0055)	**0.799** **(± 0.0019)**
	RMSE	1.198 (± 0.0105)	1.143 (± 0.0116)	1.289 (± 0.0114)	1.087 (± 0.0032)	1.233 (± 0.0163)	1.103 (± 0.0023)	1.067 (± 0.0110)	**1.038** **(± 0.0001)**
Ciao d=10	MAE	0.823 (± 0.0130)	0.768 (± 0.0108)	0.769 (± 0.0087)	0.753 (± 0.0016)	0.821 (± 0.0038)	0.745 (± 0.0012)	0.738 (± 0.0053)	**0.722** **(± 0.0019)**
	RMSE	1.088 (± 0.0099)	1.017 (± 0.0110)	1.018 (± 0.0084)	0.976 (± 0.0026)	1.082 (± 0.0134)	1.022 (± 0.0018)	0.964 (± 0.0108)	**0.955** **(± 0.0029)**

ent semantic latent spaces, i.e., different user behavior in social networks compared to recommender systems.

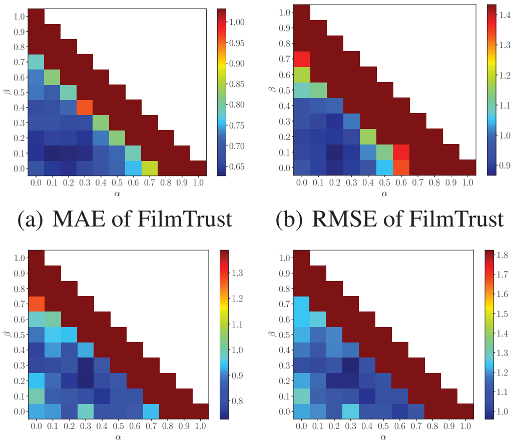
- SREPS outperforms SoDimRec which models heterogeneity and weak dependence connections in social networks as social regularization. SoDimRec models more fine-grained properties of social networks and describes user actions in more detail.

Parameter Sensitivity

In this subsection, we present the experiments conducted to further investigate the effects of hyper parameters on overall performance and provide suggestions for how to reasonably assign for their setting.

Hyper Parameters α And β The hyper parameters α and β control the influence of social networks and recommendation networks. From 0 to 1 in steps of 0.1, we experimented with different combinations of the two hyper parameters. Note that the sums of α and β should not larger than 1 due to Eq. (9). Due to space limitations, we have only presented the MAE and RMSE distributions for the FilmTrust and Ciao datasets with 5 dimensions in Fig. 3. Note that the brown-colored squares (i.e., the top color in color bars) are larger than the corresponding values. For example, the MAE in FilmTrust is nearly 2.53.

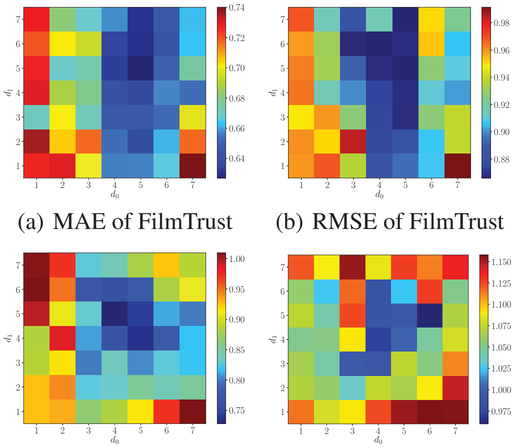
From Fig.3 we can observer that: **1)** The hyper parameter combinations near the bottom left corner (i.e., $\alpha = 0, \beta = 0$) achieved better performance, which demonstrates the positive influence of social networks and recommendation networks. **2)** The performance near the bottom right corner (i.e., $\alpha = 1, \beta = 0$) and the top left corner (i.e., $\alpha = 0, \beta = 1$) was bad. Increasing α or β decreased the contribution of rating loss. Thus, the user latent vectors in the essential preference space are mainly influenced by social networks or recommendation networks, some personalized information from the historical ratings has been omitted. **3)** A similar situation occurred near the line $\alpha + \beta = 1$. There was no rating loss in the loss function and the model did not learn any information about the rating space. The item latent vectors were dominated by the regularization, which resulted in that the item latent vectors are almost zero vectors. **4)** The SREPS model achieved the best performance at the centers of the distribution triangles. Moreover, the social networks usually contributed more than the recommendation networks. In these areas, the contributions of the three components in the loss function in Eq. (9) were balanced, and the user preferences in the rating space can be guided by the preferences in the two networks. The best α and β were different for the different datasets. However, α was usually larger than β and smaller than $1 - \alpha - \beta$, which may be helpful in selecting the best hyper parameters.



(a) MAE of FilmTrust (b) RMSE of FilmTrust

(c) MAE of Ciao (d) RMSE of Ciao

Figure 3: MAE and RMSE distributions with different combinations of α and β.



(a) MAE of FilmTrust (b) RMSE of FilmTrust

(c) MAE of Ciao (d) RMSE of Ciao

Figure 4: MAE and RMSE distributions with different combinations of d_0 and d_1.

Dimensions l, d_0, d_1 and d_2 Finer control and tuning can be achieved by assigning separate dimensions to different spaces. In this subsection, we fixed $l = 5$ and varied other dimensions. $d_0 = d_2$ are set, because both of them come from the recommender system. From 1 to 7 in steps of 1, we experimented with different combinations of d_0 and d_1. Due to space limitations, we have only presented the MAE and RMSE results of the FilmTrust and Ciao datasets in Fig. 4.

Within an appropriate dimension range, when the dimension of social embedding space (e.g., d_0) and that of rating space (i.e. d_1) is near that of essential preference space (e.g., l), the results of RMSE and MAE acceptable. When either d_0 or d_1 is far from than l, the results reduces significantly. According to the results in Fig. 4, a trivial but effective way to select these dimensions (e.g., d_0, d_1, d_2 and l) is to set all of them with the same dimensions.

Related Work

Social recommendation has been widely studied based on the latent factor models. Broadly, the following four types of methods incorporate social networks into recommender systems.

The first collective matrix factorization methods, which collectively factorize the rating matrix and social matrix, sharing the same user latent vectors. This type method captures the similarity of user preference in recommender system and social network. Ma et al. (2008) proposed a SoRec model, which shared common user latent vectors factorized by ratings and by trust. Social networks from both local and global perspectives were modeled by jointly factorizing the weighted rating matrix and social similarity matrix (Tang et al. 2013). Jamali and Lakshmanan (2013) proposed a HeteroMF model, which adapted the collective matrix factorization method into the heterogeneous information networks. Based on the social reverse height perspective, a listwise model was proposed by Rafailidis and Crestani (2016), which can be seen as a variation of the collective factorization model.

Another type of method is to modify the rating representation, i.e., a user rating can be influenced by his/her preference and neighbor preferences. The user latent vectors were linearly combined with those of the user trusted neighbors, i.e., the user ratings are balanced between his own preference and his neighbors' preference (Ma, King, and Lyu 2009). Chaney, Blei and Eliassi-Rad (2015) modified the rating as the combination of the product of latent vectors and the ratings from social neighbors under the probabilistic Poisson factorization framework.

On the other hand, the researchers consider a social networks as a regularization, which constraints that social neighbors have similar preferences. Jamali and Ester (2010) proposed that a user's latent vectors should be close to the weighted average of his social neighbors and incorporated the social network as regularization. Based on similar idea, Ma et al. (2011) proposed a individual-based social regularization, which indirectly models the propagation of tastes. Further, the heterogeneity of social relations and weak dependency connections were considered as regularizations (Tang et al. 2016).

At last, we can also use the hybrid strategy to combine the above methods. For example, Fang, Bao and Zhang (2014) jointly factorized the rating matrix and the trust matrix, and modify the trust values from the meaningful aspects of trust. Guo, Zhang and Yorker-Smith (2015) also jointly factorized the two matrices, and reformulated the ratings with the implicit effects of trusted users and historical rated items under the SVD++ framework. The strong and weak ties in social networks were modeled int PTPMF model (Wang et al. 2017). PTPMF incorporated the preferences of both strongly and weakly connected users into the rating presentation, and regularized the user latent vectors from both strong and weak tie perspectives.

Our work SREPS belongs to the collective matrix factorization method. Compared with previous works, SREPS not only captures the similarity of user preference in recommender system and social network, but also allows different preference factors in different scenarios. SREPS is more flexible to learn knowledge from social network and apply this knowledge to improve the performance of recommender system, as it was demonstrated in the experimental part.

Conclusion

In this paper, we proposed a novel social recommendation framework called SREPS, which takes the essential preference space into account to model the differences between user preferences in recommender systems and in social networks. SREPS maps the essential preference space into different semantic latent spaces using space projection matrices. By jointly incorporating rating information, consumption information and social structural information, SREPS is able to learn latent vectors in the essential preference space to produce personalized recommendations. Comprehensive experimental results on four real-world datasets demonstrate that our model provides the best performance in term of mean absolute error and root mean square error compared to seven state-of-the-art methods.

Acknowledgments

This work was supported by the NSFC (No. 61502479 and 61370025), the National Key Research and Development Program of China (No. 2016YFB0801301, 2016YFB0801003 and 2017YFB0803304), and the Youth Innovation Promotion Association CAS (No. 2017210). C. Zhou is the corresponding author.

References

Chaney, A. J.; Blei, D. M.; and Eliassi-Rad, T. 2015. A probabilistic model for using social networks in personalized item recommendation. In *RecSys*, 43–50.

Fang, H.; Bao, Y.; and Zhang, J. 2014. Leveraging decomposed trust in probabilistic matrix factorization for effective recommendation. In *AAAI*, 30–36.

Gao, L.; Wu, J.; Zhou, C.; and Hu, Y. 2017. Collaborative dynamic sparse topic regression with user profile evolution for item recommendation. In *AAAI*, 1316–1322.

Guo, G.; Zhang, J.; and Thalmann, D. 2012. A simple but effective method to incorporate trusted neighbors in recommender systems. In *UMAP*, 114–125.

Guo, G.; Zhang, J.; and Yorke-Smith, N. 2013. A novel bayesian similarity measure for recommender systems. In *IJCAI*, 2619–2625.

Guo, G.; Zhang, J.; and Yorke-Smith, N. 2015. Trustsvd: Collaborative filtering with both the explicit and implicit influence of user trust and of item ratings. In *AAAI*, 123–129.

Hsieh, C.; Yang, L.; Wei, H.; Naaman, M.; and Estrin, D. 2016. Immersive recommendation: News and event recommendations using personal digital traces. In *WWW*, 51–62.

Jamali, M., and Ester, M. 2010. A matrix factorization technique with trust propagation for recommendation in social networks. In *RecSys*, 135–142.

Jamali, M., and Lakshmanan, L. V. S. 2013. Heteromf: recommendation in heterogeneous information networks using context dependent factor models. In *WWW*, 643–654.

Koren, Y. 2008. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *KDD*, 426–434.

Krohn-Grimberghe, A.; Drumond, L.; Freudenthaler, C.; and Schmidt-Thieme, L. 2012. Multi-relational matrix factorization using bayesian personalized ranking for social network data. In *WSDM*, 173–182.

Ma, H.; Yang, H.; Lyu, M. R.; and King, I. 2008. Sorec: social recommendation using probabilistic matrix factorization. In *CIKM*, 931–940.

Ma, H.; Zhou, D.; Liu, C.; Lyu, M. R.; and King, I. 2011. Recommender systems with social regularization. In *WSDM*, 287–296.

Ma, H.; King, I.; and Lyu, M. R. 2009. Learning to recommend with social trust ensemble. In *SIGIR*, 203–210.

Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G. S.; and Dean, J. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS*, 3111–3119.

Ohsawa, S.; Obara, Y.; and Osogami, T. 2016. Gated probabilistic matrix factorization: Learning users' attention from missing values. In *IJCAI*, 1888–1894.

Rafailidis, D., and Crestani, F. 2016. Joint collaborative ranking with social relationships in top-n recommendation. In *CIKM*, 1393–1402.

Salakhutdinov, R., and Mnih, A. 2007. Probabilistic matrix factorization. In *NIPS*, 1257–1264.

Tang, J.; Gao, H.; Liu, H.; and Sarma, A. D. 2012. etrust: understanding trust evolution in an online world. In *KDD*, 253–261.

Tang, J.; Hu, X.; Gao, H.; and Liu, H. 2013. Exploiting local and global social context for recommendation. In *IJCAI*, 2712–2718.

Tang, J.; Qu, M.; Wang, M.; Zhang, M.; Yan, J.; and Mei, Q. 2015. LINE: large-scale information network embedding. In *WWW*, 1067–1077.

Tang, J.; Wang, S.; Hu, X.; Yin, D.; Bi, Y.; Chang, Y.; and Liu, H. 2016. Recommendation with social dimensions. In *AAAI*, 251–257.

Tang, J.; Gao, H.; and Liu, H. 2012. mtrust: discerning multi-faceted trust in a connected world. In *WSDM*, 93–102.

Wang, X.; Hoi, S. C. H.; Ester, M.; Bu, J.; and Chen, C. 2017. Learning personalized preference of strong and weak ties for social recommendation. In *WWW*, 1601–1610.

Yang, B.; Lei, Y.; Liu, D.; and Liu, J. 2013. Social collaborative filtering by trust. In *IJCAI*, 2747–2753.