An Experimental Study on Implicit Social Recommendation

Hao Ma
Microsoft Research
One Microsoft Way
Redmond, WA 98052
haoma@microsoft.com

ABSTRACT

Social recommendation problems have drawn a lot of attention recently due to the prevalence of social networking sites. The experiments in previous literature suggest that social information is very effective in improving traditional recommendation algorithms. However, explicit social information is not always available in most of the recommender systems, which limits the impact of social recommendation techniques. In this paper, we study the following two research problems: (1) In some systems without explicit social information, can we still improve recommender systems using implicit social information? (2) In the systems with explicit social information, can the performance of using implicit social information outperform that of using explicit social information? In order to answer these two questions, we conduct comprehensive experimental analysis on three recommendation datasets. The result indicates that: (1) Implicit user and item social information, including similar and dissimilar relationships, can be employed to improve traditional recommendation methods. (2) When comparing implicit social information with explicit social information, the performance of using implicit information is slightly worse. This study provides additional insights to social recommendation techniques, and also greatly widens the utility and spreads the impact of previous and upcoming social recommendation approaches.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Information Filtering

General Terms

Algorithm, Experimentation

Keywords

Matrix Factorization, Recommender Systems, Implicit Social Information, Singular Value Decomposition

1. INTRODUCTION

Due to the rapid growth of information on the Web, especially on the social Web, recommender system has become an indispensable technique for filtering and recommending online information. In order to satisfy Web users' ever-increasing information needs, traditional recommendation techniques have been widely adopted by various products in industrial companies, including but not limited to Amazon, Netflix, Apple iTunes, Yahoo! News, etc.

Traditional recommendation techniques normally only take into account the user-item rating matrix for computing recommendations. Recently, based on the intuition that users' social network information can be utilized to improve recommendation qualities, the research of social recommender systems becomes popular. Several social recommendation approaches [8, 9, 20, 21] have been proposed in the literature. These methods suggest that the explicit social information is very helpful in improving the traditional methods, especially when the user-item rating matrix is sparse.

In general, most of the social recommendation methods are based on the matrix factorization framework, which is both effective and efficient in generating recommendations. Typically, social information is utilized to better shape the user latent space. Different intuitions on interpreting these social information will result in different objective functions or learning models. In [20], a social recommendation approach is proposed by adding the social regularization term to the matrix factorization objective function. In this method, the additional social regularization term ensures that the distance of the latent feature vectors of two friends will become closer if these two friends share similar tastes. The experimental results illustrate that social recommendation approaches are very effective at improving traditional recommendation techniques, especially when few ratings are available.

Although social recommendation methods have been extensively studied in the literature, many problems are still left unexplored.

The essence of social recommendation methods is utilizing users' explicit social connections to improve recommendation results. However, explicit social connection information is not always available in real-world recommender systems. Only few Web sites have implemented the social or trust mechanisms, like Epinions (http://www.epinions.com, a general consumer review site that was established in 1999, where users can also add other users into their trust list) and Doudan (http://www.douban.com, the largest Chinese Web 2.0 site devoted for movies, books, and music reviews
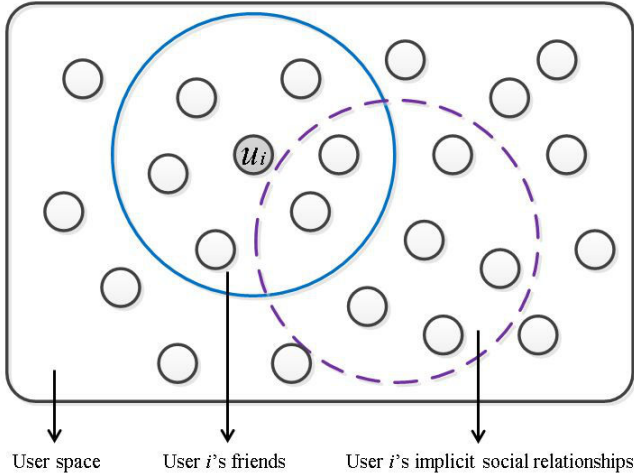
Figure 1: Explicit and Implicit Social Relationships

that was launched in 2005). Lacking of social recommendation data greatly limits the impact and utilization of social recommendation methods.

Fortunately, in case that we do not have explicit social information, we can always compute a set of implicit social information for each user. We summarize the relationships between users' explicit social information and implicit social information in a toy example illustrated in Figure 1. In this figure, users within the blue solid circle represent explicit social connections of user u_i, while users within the purple dashed circle represent this user's implicit social information, which is calculated using some similarity metrics.

Thus, in this paper, based on the example shown in Figure 1, we are interested in exploring the following two research problems:

1. Can we take advantages of implicit social information in case we do not have explicit social connection information?

2. When the explicit social information is available, can the performance of using implicit social information outperform that of using explicit social information?

In order to answer the questions, in this paper, we conduct comprehensive experiments on three datasets: the Movie-Lens and EachMovie datasets, which do not have explicit social information, as well as the Douban dataset, which has a social network in addition to the user-item rating matrix. Based on our analysis, we have the following conclusions:

- When explicit social information is **unavailable**:

  - In the absence of *explicit* user social information, we find that using *implicit* user social information (i.e., the most similar users), can also improve the recommendation qualities under the matrix factorization framework.
  - Besides the most *similar* users, for each user, the list of most *dissimilar* users is another ideal source to further improve recommender systems.
  - We extend the idea of computing implicit *user* social information, and conclude that the implicit

item social information is also very helpful in increasing the recommendation performance.

- When explicit social information is **available**:

  - We find that using implicit social information performs slightly worse than using explicit social information.
  - We also conclude that a social network with larger friend diversity is more effective in improving recommendation quality.

The remainder of this paper is organized as follows. Section 2 introduces previous methods that are related to this work. Section 3 details one popular matrix factorization method in the literature. Section 4 presents how to incorporate implicit social information. The results of an empirical analysis are presented in Section 5, followed by the conclusion in Section 6.

2. RELATED WORK

In this section, we review several popular approaches for recommender systems in the literature, including: (1) traditional recommender systems, especially matrix factorization based methods, and (2) social recommendation methods.

2.1 Traditional Recommender Systems

Traditional recommender systems normally only utilize the user-item rating information for recommendation. One of the the most popular techniques in recommender systems is collaborative filtering.

Typically, collaborative filtering approaches include two types of methods: memory-based methods as well as model-based approaches. Memory-based methods focus on using predefined similarity calculation functions to find similar users or items for generating predictions. Memory-based methods can be further classified as user-based [2, 4, 10] and item-based approaches [3, 16, 27] based on whether similar users or similar items are used.

In contrast to the memory-based methods, the model-based approaches use the observed ratings to train a predefined learning model. The ratings are then predicted via the trained model instead of directly manipulating the original rating database as the memory-based approaches [17]. Algorithms in this category include but not limited to clustering model [11], the aspect models [5, 6, 28], the Bayesian hierarchical model [33], the ranking model [17], etc. Hofmann in [5] proposed an algorithm based on a generalization of probabilistic latent semantic analysis to continuous-valued response variables. Kohrs et al. [11] presented an algorithm for collaborative filtering based on hierarchical clustering, which tried to balance both robustness and accuracy of predictions, especially when few data were available. Recently, due to the effectiveness and efficiency in dealing with very large user-item rating matrices, several low-dimensional matrix factorization techniques [15, 24, 25, 26, 29, 31] have been proposed.

Matrix factorization methods in recommender systems normally seek to factorize the user-item rating matrix into two low rank user-specific and item-specific matrices, and then utilize the factorized matrices to make further predictions. Low-rank matrix approximations based on minimizing the sum-squared errors can be easily solved using Singular Value

Decomposition (SVD), and a simple and efficient Expectation Maximization (EM) algorithm for solving weighted low-rank approximation is proposed in [29]. In [30], Srebro et al. proposed a matrix factorization method to constrain the norms of U and V instead of their dimensionality. Salakhutdinov et al. presented a probabilistic linear model with Gaussian observation noise in [25]. The user and item latent factors can be learned by maximize the proposed probabilistic likelyhood function. The proposed method is very effective and efficient, and this method is essentially equivalent to the Regularized SVD method. In their following work proposed in [26], Salakhutdinov et al. placed the Gaussian-Wishart priors on the user and item hyperparameters, which can further improve the prediction accuracy. More recently, Koren et al. [12, 13, 14] proposed several enhanced matrix factorization methods which illustrate promising results by incorporating heterogeneous information. In [12], the authors discussed the possibility to improve recommender systems using neighborhood information, which is quite related to our work. However, in this paper, we focus on very different aspect of recommender system, i.e., social recommendation. Moreover, we provide many additional insights that previous work ignored.

2.2 Social Recommendation

The above mentioned traditional recommendation techniques are all based on working on the user-item rating matrix, and ignore the abundant relationships among users. Recently, due to the prevalence of Web 2.0 social networking sites, many researchers have started studying social recommender systems [1, 7, 8, 9, 20, 21, 22, 23, 32].

Due to the nature of each method, the social recommendation techniques can also be classified into two types: memory-based [1, 8, 21, 22, 23] and model-based [7, 9, 20, 32].

The memory-based methods normally directly or indirectly use the degree of social trust to represent the similarity between two users. In [21], a trust-aware method for recommender system is proposed. In this work, the collaborative filtering process is replaced by the reputation of users, which is computed by propagating trust. The degrees of trust are calculated to replace the similarity value between two users. The experiments on a large social recommendation dataset - Epinions, show that this work increases the coverage (number of ratings that are predictable) while not decreasing the accuracy (the error of predictions). In [23], two trust-aware methods are proposed to improve standard collaborative filtering methods. The experimental results indicate that the user trust information can help improve recommendation quality.

More recently, by taking advantages of the effectiveness and efficiency of matrix factorization framework, several novel model-based methods have been proposed to enhance traditional matrix factorization methods by incorporating user social information. In [20], two Social Regularization methods have been proposed by constraining the matrix factorization objective function with user social regularization terms. Different with previous methods, the proposed methods are very general, they not only work with user trust relationships, but also perform well with user social friend relationships. The experimental analysis indicates that the proposed framework outperforms other state-of-the-art methods.

Although the aforementioned matrix factorization based social recommendation methods move a nice step forward

in the research of recommender systems, their utilities are limited since these methods require using explicit user social information, which is not always available in most of the recommender systems. In [18], the idea of using implicit information is briefly introduced, however, many interesting problems are left unattended. In this paper, we present an comprehensive experimental study on implicit social recommendation which provides many useful insights to current social recommendation techniques.

3. MATRIX FACTORIZATION

In this subsection, we review one popular matrix factorization method that is widely studied in the literature.

Considering an $m \times n$ matrix R describing m users' ratings on n items, a low-rank matrix factorization approach seeks to approximate the frequency matrix R by a multiplication of d-rank factors $R \approx U^T V$, where $U \in \mathbb{R}^{d \times m}$ and $V \in \mathbb{R}^{d \times n}$ with $d << \min(m, n)$. The matrix R in the real-world is usually very sparse since most of the users only visited a few Web sites.

Traditionally, the Regularized Singular Value Decomposition (RSVD) method is employed to estimate a matrix R by minimizing

$$
\begin{aligned}
L \;=\; & \min_{U,V} \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{n} I_{ij}(r_{ij} - \mathbf{u}_i^T \mathbf{v}_j)^2 \\
& + \frac{\lambda_1}{2} ||U||_F^2 + \frac{\lambda_2}{2} ||V||_F^2,
\end{aligned}
\tag{1}
$$

where $\mathbf{u}_i$ and $\mathbf{v}_j$ are column vectors with d values, I_{ij} is the indicator function that is equal to 1 if user i rated item j and equal to 0 otherwise, and λ_1, λ_2 represent the regularization parameters. The optimization problem in Equation 1 minimizes the sum-of-squared-errors objective function with quadratic regularization terms.

By adopting a simple stochastic gradient descent technique, for each observed rating r_{ij}, we have the following efficient updating rules to learn latent variables $\mathbf{u}_i$, $\mathbf{v}_j$:

$$
\begin{aligned}
\mathbf{u}_i &\leftarrow \mathbf{u}_i + \gamma_1(\Delta_{ij}\mathbf{v}_j - \lambda_1 \mathbf{u}_i), \\
\mathbf{v}_j &\leftarrow \mathbf{v}_j + \gamma_2(\Delta_{ij}\mathbf{u}_i - \lambda_2 \mathbf{v}_j),
\end{aligned}
\tag{2}
$$

where

$$
\Delta_{ij} = r_{ij} - \mathbf{u}_i^T \mathbf{v}_j,
$$

and γ_1, γ_2 are the learning rates.

The Regularized SVD algorithm introduced in this section is both effective and efficient in solving the collaborative filtering problem and it is perhaps one of the most popular methods in collaborative filtering. In this paper, we use this approach as the baseline method.

4. INCORPORATING IMPLICIT SOCIAL INFORMATION

In this section, we first introduce a matrix factorization-based social recommendation method proposed in the literature. Then we illustrate how to leverage the implicit user social information in the case that explicit user social information is not available. Finally, we demonstrate how to utilize dissimilar users as well as item social information to further improve recommender systems.

Note that the main focus of this paper is designing a scientific experimental study to explore several interesting research problems by borrowing and extending previous social

recommendation techniques. Developing brand new social recommendation techniques is out of the scope of this paper. Also, we will only briefly describe the techniques if they are borrowed from previous work.

4.1 Social Regularization

In [20], in order to model the social recommendation problems more accurately, a general social recommendation approach, Social Regularization (SR), is proposed.

The objective function of this approach is formulated as:

$$
\begin{aligned}
L \quad = \quad & \min_{U,V} \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{n} I_{ij}(r_{ij} - \mathbf{u}_i^T \mathbf{v}_j)^2 \\
& + \frac{\alpha}{2} \sum_{i=1}^{m} \sum_{f \in \mathcal{F}^+(i)} s_{if} \|\mathbf{u}_i - \mathbf{u}_f\|_F^2 \\
& + \frac{\lambda_1}{2} \|U\|_F^2 + \frac{\lambda_2}{2} \|V\|_F^2,
\end{aligned} \quad (3)
$$

where α is the regularization parameter, s_{if} indicates the similarity between user i and user f, and $\mathcal{F}^+(i)$ represents user i's outlink friends.

In this method, the social network information is employed in designing the social regularization term to constrain the matrix factorization objective function. The social regularization term also indirectly models the propagation of tastes. More specifically, if user i has a friend f and user f has a friend user g, this regularization term actually indirectly minimizes the distance between latent vectors $\mathbf{u}_i$ and $\mathbf{u}_g$. The propagation of tastes will reach a harmonic status once the learning is converged.

Similarly, for each observed rating r_{ij}, we have the following stochastic gradient descent updating rules to learn the latent parameters:

$$
\begin{aligned}
\mathbf{u}_i \quad \leftarrow \quad & \mathbf{u}_i + \gamma_1 \Big(\Delta_{ij} \mathbf{v}_j - \alpha \sum_{f \in \mathcal{F}^+(i)} s_{if}(\mathbf{u}_i - \mathbf{u}_f) \\
& - \alpha \sum_{g \in \mathcal{F}^-(i)} s_{ig}(\mathbf{u}_i - \mathbf{u}_g) - \lambda_1 \mathbf{u}_i \Big), \\
\mathbf{v}_j \quad \leftarrow \quad & \mathbf{v}_j + \gamma_2 (\Delta_{ij} \mathbf{u}_i - \lambda_2 \mathbf{v}_j), \quad (4)
\end{aligned}
$$

where

$$
\Delta_{ij} = r_{ij} - \mathbf{u}_i^T \mathbf{v}_j, \quad (5)
$$

and $\mathcal{F}^-(i)$ represents user i's inlink friends.

4.2 Implicit User Social Relationships

As mentioned in Section 1, all the social recommendation approaches need to utilize the additional explicit user social information, which may limit the impact and utilization of these approaches. In this section, we seek an alterative way to compute implicit user social information once the explicit user social relationships are unavailable.

The essence of social recommendation approaches lies in the additional explicit social information of each user. The information of these social friends can then be utilized to help model a user's taste more accurately. In the case of missing explicit social information, as shown in Figure 1, we can always compute another set of Top-N similar users and then plug in those similar users to the aforementioned social recommendation matrix factorization framework.

There are several methods we can borrow in the literature to compare the similarity between two users. In this paper,

we adopt the most popular approach Pearson Correlation Coefficient (PCC) [2], which is defined as:

$$
s_{if} = \frac{\displaystyle\sum_{k \in I(i) \cap I(f)} (r_{ik} - \overline{r}_i) \cdot (r_{fk} - \overline{r}_f)}{\sqrt{\displaystyle\sum_{k \in I(i) \cap I(f)} (r_{ik} - \overline{r}_i)^2} \cdot \sqrt{\displaystyle\sum_{k \in I(i) \cap I(f)} (r_{fk} - \overline{r}_f)^2}}, \quad (6)
$$

where $I(i)$ is a set of items that rated by user i, and $\overline{r}_i$ represents the average rate of user i. From this definition, user similarity s_{if} is ranging from $[-1, 1]$, and a larger value means users i and f are more similar. We employ a mapping function $f(x) = (x+1)/2$ to bound the range of PCC similarities into $[0, 1]$.

Based on the PCC similarity, the computed Top-N similar users can then be injected into the objective function detailed in Equation 3.

4.3 Dissimilar Users

So far in this paper, we only consider utilizing similar users to improve recommender systems. In the social regularization term $s_{if} \|u_i - u_f\|_F^2$ employed in Equation 3, if user i is similar to user f, the distance between latent vectors u_i and u_f will become closer since the similarity s_{if} is a relatively large value.

Motivated by the similar users, we can actually endow the social regularization term more modeling power if we also include the most dissimilar users. If user i is dissimilar with user f, the ideal property we want is to make the distance between u_i and u_f larger or in other words, maximize the distance between them. Inspired by the work in [19], we can easily achieve this property by turning the similarity $s_{if} = -s_{if}$. Hence, in this case, the social regularization term $s_{if} \|u_i - u_f\|_F^2$ will actually lead to make the distance between u_i and u_f larger. This is an ideal property we desire to include those dissimilar users.

Hence, in this paper, in addition to the Top-N similar users, we also include the Top-N dissimilar users for each user. This will not change the objective function mentioned in Equation 3. It will only increase the size of the implicit social neighbors specified in the set $\mathcal{F}^+(i)$ and change the signs of the similarity values for dissimilar users.

4.4 Item Social Relationships

In the original social recommendation problems, there are only social relationships among users due to the reason that normally, social network only refers to the social relationships between people.

In this paper, since we define the implicit user social information as the similar or dissimilar users, we can naturally extend this idea to also take advantages of the implicit item social information, which can be found through the similar or dissimilar items.

The Social Regularization method described in Section 4.1 is a very general approach, and it can be easily extended to incorporate the item social information. The objective function can be formulated as:

$$L = \min_{U,V} \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{n}I_{ij}(r_{ij} - \mathbf{u}_i^T\mathbf{v}_j)^2$$
$$+\frac{\beta}{2}\sum_{j=1}^{n}\sum_{q\in\mathcal{Q}^+(j)}s_{jq}\|\mathbf{v}_j - \mathbf{v}_q\|_F^2$$
$$+\frac{\lambda_1}{2}\|U\|_F^2 + \frac{\lambda_2}{2}\|V\|_F^2, \qquad (7)$$

where $Q^+(j)$ represents item j's implicit social information (i.e., the Top-N items similar to item j and the Top-N items dissimilar to item j).

The similarity between item j and item q can be calculated by the item-based PCC method:

$$s_{jq}=\frac{\displaystyle\sum_{k\in U(j)\cap U(q)}(r_{kj}-\overline{r}_j)\cdot(r_{kq}-\overline{r}_q)}{\sqrt{\displaystyle\sum_{k\in U(j)\cap U(q)}(r_{kj}-\overline{r}_j)^2}\cdot\sqrt{\displaystyle\sum_{k\in U(j)\cap U(q)}(r_{kq}-\overline{r}_q)^2}}, \quad (8)$$

where $U(j)$ denotes a set of users that rated item j.

Similarly, we have the following updating rules to learn the latent parameters:

$$\mathbf{u}_i \leftarrow \mathbf{u}_i + \gamma_1(\Delta_{ij}\mathbf{v}_j - \lambda_1\mathbf{u}_i),$$
$$\mathbf{v}_j \leftarrow \mathbf{v}_j + \gamma_2\Big(\Delta_{ij}\mathbf{u}_i - \beta\sum_{q\in\mathcal{Q}^+(j)}s_{jq}(\mathbf{v}_j - \mathbf{v}_q)$$
$$-\beta\sum_{h\in\mathcal{Q}^-(j)}s_{jh}(\mathbf{v}_j - \mathbf{v}_h) - \lambda_1\mathbf{v}_j\Big), \qquad (9)$$

where

$$\Delta_{ij} = r_{ij} - \mathbf{u}_i^T\mathbf{v}_j. \qquad (10)$$

4.5 A Unified Model

From Section 4.2 to Section 4.4, we demonstrate how to utilize implicit user social information, dissimilar users, and item social information, respectively. We can then design the following integrated model to take into account all the possible information that will potentially benefit the recommender systems:

$$L = \min_{U,V} \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{n}I_{ij}(r_{ij} - \mathbf{u}_i^T\mathbf{v}_j)^2$$
$$+\frac{\alpha}{2}\sum_{i=1}^{m}\sum_{f\in\mathcal{F}^+(i)}s_{if}\|\mathbf{u}_i - \mathbf{u}_f\|_F^2$$
$$+\frac{\beta}{2}\sum_{j=1}^{n}\sum_{q\in\mathcal{Q}^+(j)}s_{jq}\|\mathbf{v}_j - \mathbf{v}_q\|_F^2$$
$$+\frac{\lambda_1}{2}\|U\|_F^2 + \frac{\lambda_2}{2}\|V\|_F^2, \qquad (11)$$

For every observed rating r_{ij}, we have the following stochastic updating rules to learn the all the latent parameters:

$$\mathbf{u}_i \leftarrow \mathbf{u}_i + \gamma_1\Big(\Delta_{ij}\mathbf{v}_j - \alpha\sum_{f\in\mathcal{F}^+(i)}s_{if}(\mathbf{u}_i - \mathbf{u}_f)$$
$$-\alpha\sum_{g\in\mathcal{F}^-(i)}s_{ig}(\mathbf{u}_i - \mathbf{u}_g) - \lambda_1\mathbf{u}_i\Big),$$
$$\mathbf{v}_j \leftarrow \mathbf{v}_j + \gamma_2\Big(\Delta_{ij}\mathbf{u}_i - \beta\sum_{q\in\mathcal{Q}^+(j)}s_{jq}(\mathbf{v}_j - \mathbf{v}_q)$$
$$-\beta\sum_{h\in\mathcal{Q}^-(j)}s_{jh}(\mathbf{v}_j - \mathbf{v}_h) - \lambda_1\mathbf{v}_j\Big), \qquad (12)$$

where

$$\Delta_{ij} = r_{ij} - \mathbf{u}_i^T\mathbf{v}_j.$$

The unified model is constrained by four types of information: similar user regularization, dissimilar user regularization, similar item regularization and dissimilar item regularization. We use the aforementioned information to help better shape the user and item latent spaces, hence generate more accurate recommendation results.

5. EXPERIMENTAL ANALYSIS

In this section, we conduct several experiments to compare different recommendation methods using implicit social information. Our experiments are intended to address the following questions:

- When explicit social information is **unavailable**:

  1. Is implicit user social information effective in improving traditional matrix factorization methods?
  2. Can dissimilar users be used to further improve the recommendation quality?
  3. Can we also take advantages of implicit item social information in addition to the implicit user social information?
  4. What is the performance comparison on users with different observed ratings?

- When explicit social information is **available**:

  1. Can the performance of using implicit social information outperform that of using explicit social information?

We use the popular Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) metrics to measure the prediction quality of all the mentioned algorithms. MAE is defined as:

$$MAE = \frac{\sum_{i,j}|r_{ij} - \widehat{r}_{ij}|}{N}, \qquad (13)$$

where r_{ij} denotes the rating user i gave to item j, $\widehat{r}_{ij}$ denotes the related predicted rating, and N denotes the number of tested ratings. RMSE is defined as:

$$RMSE = \sqrt{\frac{\sum_{i,j}(r_{ij} - \widehat{r}_{ij})^2}{N}}. \qquad (14)$$

From the definitions, we can see that a smaller MAE or RMSE indicates a better performance.

Table 1: Statistics of Dataset MovieLens

Statistics	User	Item
Min. Num. of Ratings	20	1
Max. Num. of Ratings	737	583
Avg. Num. of Ratings	106.04	59.45

Table 2: Statistics of Dataset EachMovie

Statistics	User	Item
Min. Num. of Ratings	1	1
Max. Num. of Ratings	1,455	32,868
Avg. Num. of Ratings	37.78	1,706.30

5.1 Without Explicit Social Information

5.1.1 Description of Datasets

When the explicit social information is not available, we evaluate all the algorithms on two popular datasets: MovieLens[1] and EachMovie[2].

The MovieLens dataset we adopt in this paper is a relatively small dataset contains 100,000 user-item ratings (scale from 1 to 5) rated by 943 users on 1,642 items. The EachMovie data set is a relatively large dataset includes 74,424 users, 1,648 movies, and 2,811,718 ratings in the range from 0 to 5.

Other statistics of these two datasets are summarized in Table 1 and Table 2, respectively.

5.1.2 Performance Analysis

In this section, we will compare the following different methods described in this paper as well as some baseline methods.

1. **UserMean**: this is a baseline method uses the mean value of every user to predict the missing values.

2. **ItemMean**: this is a baseline method utilizes the mean value of every item to predict the missing values.

3. **RSVD**: this is the Regularized SVD method. It is equivalent with the method proposed by Salakhutdinov and Minh in [25]. The underlining distribution is assumed as Gaussian distribution. The details of this method are also introduced in Section 3.

4. **SR^{u+}**: this is the Social Regularization method using implicit similar user information. The notation u in the superscript denotes that implicit user social information is used in this method, while the notation "+" indicates only similar users are included.

5. **SR^{u+-}**: this is the Social Regularization method using both implicit similar and dissimilar user information. The notation "−" indicates dissimilar information is included.

6. **SR_{i+}**: this is the Social Regularization method using implicit similar item information. We use the subscript to describe item related information.

7. **SR_{i+-}**: this is the Social Regularization method using both implicit similar and dissimilar item information.

8. **SR_{i+-}^{u+-}**: this is the final integrated model that using similar and dissimilar user information as well as similar and dissimilar item information.

Data Preparation.

For all the experiments conducted in this section, we utilize 80% as training data in both datasets. Training data 80%, for example, means we randomly select 80% of the ratings from the MovieLens or the EachMovie dataset as the training data to predict the remaining 20% of ratings.

Top-N Neighbors Generation.

The methods we study in this paper also involve the calculation of the Top-N similar and Top-N dissimilar users or items. We adopt the following rules to generate Top-N similar and dissimilar users or items. In order to reduce the noises when computing the similarities using PCC method between two users i and f, we require that user i and user f should at least co-rated 10 items, otherwise, we will ignore user f when computing user i's Top-N similar or dissimilar neighbors, and vice versa. Furthermore, for all the similar neighbors, the similarity between two users should be greater than 0.75, while for all the dissimilar neighbors, the similarity between two users should be less than 0.25. The same rules are also adopted when calculating similar and dissimilar items.

Parameter Settings.

In order to fairly compare every method, we employ similar parameter settings for those common parameters adopted in all the approaches. In this paper, for RSVD, SR^{u+}, SR^{u+-}, SR_{i+}, SR_{i+-} and SR_{i+-}^{u+-}, we use the setting $\lambda_1 = \lambda_2 = 0.01$. At the same time, all the learning rates γ_1 and γ_2 are set to 0.005. For all the Social Regularization based methods, α and β are set to 0.015 in the MovieLens dataset, while in the EachMovie dataset, they are set to 0.001.

Performance Analysis.

The experimental results using 10 and 50 dimensions to represent the latent factors in two different datasets are shown in Table 3 and Table 4, respectively. The percentages in the results are the improvements of our SR_{i+-}^{u+-} method over the corresponding approaches. In all our methods, the number of implicit user or item social neighbors are set to 10.

The following summarizes the key conclusions we observe from the results:

- We first notice that approach SR^{u+} outperforms the RSVD method, which only utilizes the user-item rating matrix. This observation coincides with our intuition that, at the absence of the explicit user social network, employing implicit user social information can help increase the recommendation quality.

- Secondly, besides the implicit user social information, the implicit item social information can also be used to improve the recommendation quality, as demonstrated by the method SR_{i+}. Among SR^{u+} and SR_{i+}, we observe that SR_{i+} generates better results than SR^{u+} in both datasets. A possible reason is that the rating styles of items typically have less diversities that those of users. Hence, the similarity calculation based on

Table 3: Performance Comparisons (MovieLens)

Dataset	Dimension	Metrics	UserMean	ItemMean	RSVD	SR^{u+}	SR^{u+-}	SR_{i+}	SR_{i+-}	SR^{u+-}_{i+-}
MovieLens	10D	MAE	0.8389	0.8274	0.7525	0.7411	0.7400	0.7409	0.7398	**0.7333**
		Improve	12.59%	11.37%	2.55%					
		RMSE	1.0466	1.0359	0.9540	0.9421	0.9406	0.9403	0.9391	**0.9284**
		Improve	11.29%	10.37%	2.68%					
	50D	MAE	0.8389	0.8274	0.7416	0.7330	0.7321	0.7329	0.7320	**0.7281**
		Improve	13.21%	12.00%	1.82%					
		RMSE	1.0466	1.0359	0.9413	0.9298	0.9291	0.9292	0.9280	**0.9221**
		Improve	11.90%	10.99%	2.04%					

Table 4: Performance Comparisons (EachMovie)

Dataset	Dimension	Metrics	UserMean	ItemMean	RSVD	SR^{u+}	SR^{u+-}	SR_{i+}	SR_{i+-}	SR^{u+-}_{i+-}
EachMovie	10D	MAE	1.1409	1.1020	0.8854	0.8737	0.8725	0.8701	0.8690	**0.8681**
		Improve	23.91%	21.23%	1.95%					
		RMSE	1.4278	1.3851	1.1678	1.1520	1.1506	1.1491	1.1487	**1.1440**
		Improve	19.88%	17.41%	2.04%					
	50D	MAE	1.1409	1.1020	0.8751	0.8631	0.8614	0.8597	0.8588	**0.8571**
		Improve	24.88%	22.22%	2.06%					
		RMSE	1.4278	1.3851	1.1579	1.1413	1.1401	1.1389	1.1374	**1.1333**
		Improve	20.63%	18.18%	2.12%					

(a) MAE Comparison on Different User Rating Scales (MovieLens)

(b) RMSE Comparison on Different User Rating Scales (MovieLens)

(c) Distribution of Testing Data (MovieLens)
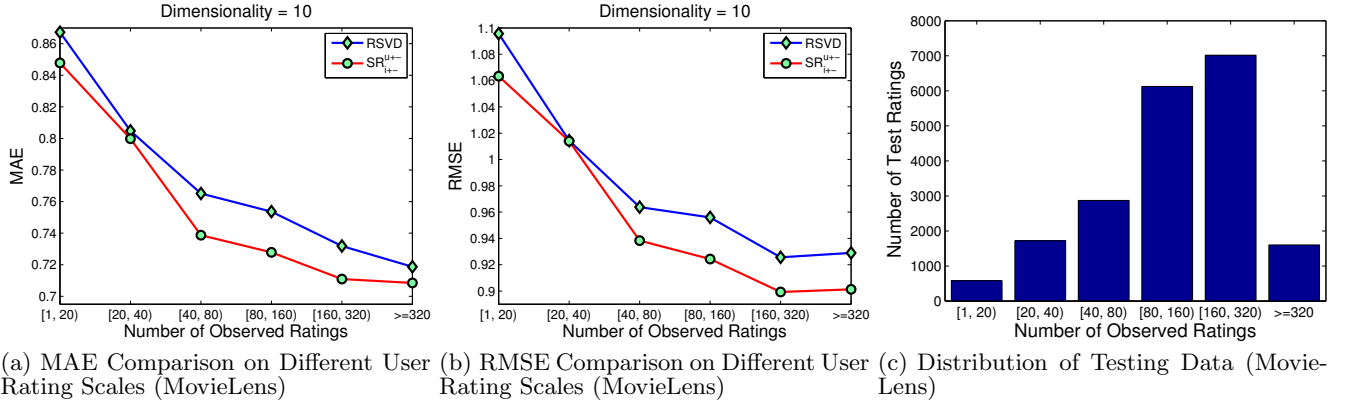
Figure 2: Performance Comparison on Different Users (MovieLens)

(a) MAE Comparison on Different User Rating Scales (EachMovie)

(b) RMSE Comparison on Different User Rating Scales (EachMovie)

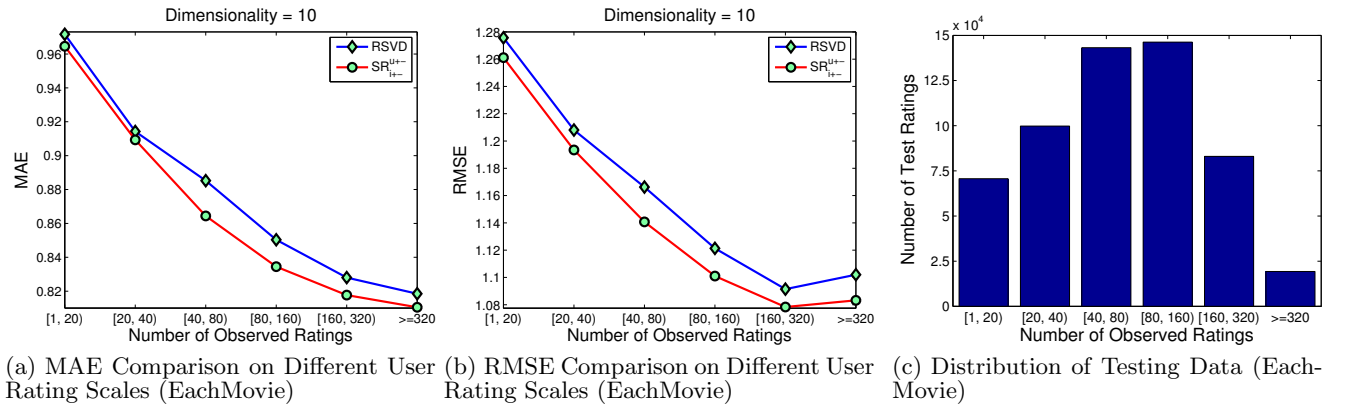(c) Distribution of Testing Data (EachMovie)

Figure 3: Performance Comparison on Different Users (EachMovie)

items is probably more accurate than the calculation based on users.

- Thirdly, another key observation we find through the experiments is that dissimilar user or item information can be used to further improve the recommender systems, as presented by the approaches SR^{u+-} and SR_{i+-}. They generate slightly better results than SR^{u+} and SR_{i+}, respectively.

- Fourthly, an integrated model SR_{i+-}^{u+-} demonstrates the best performance by incorporating all the useful implicit social information, including similar and dissimilar users as well as similar and dissimilar items. In general, the experimental results not only prove the effectiveness of incorporating implicit social information, but also demonstrate the flexibility of social regularization framework.

5.1.3 Prediction Accuracy on Different Users

In order to analyze the experiments thoroughly, in this section, 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 in the training datasets based on the number of observed ratings, and then measure the prediction accuracies of different user groups. The experimental results conducted in both Movie-Lens and EachMovie datasets are illustrated in Figure 2 and Figure 3, respectively. In these two figures, in order to interpret the results more intuitively, we include the baseline method RSVD for comparison since it does not include any social information.

Users are grouped into 6 classes: "[1, 20)", "[20, 40)", "[40, 80)", "[80, 160)","[160, 320)" and ">=320". Figure 2(c) and Figure 3(c) summarizes the distributions of the number of testing data according to the groups in the training data. For example, in the EachMovie dataset, there are a total of 70,677 user-item pairs need to be predicted in the testing dataset in which the related users in the training dataset have rated 1 to 19 items.

From Figure 2(a), Figure 2(b), Figure 3(a) and Figure 3(b), we can see that the method SR_{i+-}^{u+-} with implicit social information consistently outperforms the RSVD method in all the user groups. We also notice an interesting phenomenon, that is, the method SR_{i+-}^{u+-} performs much better than RSVD when more ratings are observed. Actually, when more ratings are observed for a user, the similarity calculation process will find more accurate similar or dissimilar neighbors for this user since we have more information to represent or interpret this user. Hence, it will perform better than RSVD especially when more ratings are observed.

5.2 With Explicit Social Information

5.2.1 Description of Dataset

We use the Douban[3] dataset in this subsection since in addition to the user-item rating matrix, it also contains a social friend network between users.

Douban is a Chinese Web 2.0 Web site providing user rating, review and recommendation services for movies, books and music. Users can assign 5-scale integer ratings (from 1 to 5) to movies, books and music. It also provides Facebook-like social networking services, which allows users to find

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

Table 5: Statistics of User-Item Matrix of Douban

Statistics	User	Item
Min. Num. of Ratings	1	1
Max. Num. of Ratings	6,328	49,504
Avg. Num. of Ratings	129.98	287.51

Table 6: Statistics of Friend Network of Douban

Statistics	Friends per User
Max. Num.	986
Avg. Num.	13.07

Table 7: Performance Comparisons (5D)

Training Data	Metrics	SR_{exp}	SR_{imp}	SR_{imp}^{top10}
40%	MAE	**0.5698**	0.5707	0.5705
	RMSE	**0.7214**	0.7223	0.7222
60%	MAE	**0.5640**	0.5648	0.5645
	RMSE	**0.7133**	0.7142	0.7139

their friends through their email accounts. This means that most of the friends on Douban actually know each other offline.

The Douban dataset we study in this paper contains 129,490 unique users and 58,541 unique movies with 16,830,839 movie ratings. As to the social friend network, the total number of friend links between users is 1,692,952. The statistics of the Douban user-item rating matrix and social friend network are summarized in Table 5 and Table 6, respectively.

5.2.2 Performance Analysis

We compare the following three methods using Douban datasets:

1. **SR_{exp}**: this is the social regularization method described in Equation 3, which utilizes the explicit social information in improving recommender systems.

2. **SR_{imp}**: this is the social regularization method that uses the implicit social information. Suppose that user u_i has n explicit social connections in the Douban dataset, then we will choose the most similar n users as the implicit social connections in this method. This setting is employed to fairly compare the method SR_{imp} with SR_{exp}.

3. **SR_{imp}^{top10}**: this is also a social regularization method uses implicit social information. Different with SR_{imp}, for each user, we use the top-10 most similar users as the implicit connections.

All the parameters of the above three methods are identical for fairly comparison.

The results are summarized in Table 7. In this table, we evaluate three methods using different percentages of training data, i.e., 40% and 60%. From the results, we surprisedly find that for all the settings, the SR_{exp} method performs slightly better than both SR_{imp} and SR_{imp}^{top10} approaches, which indicates that in recommender systems, using user-established explicit social connections are better than computer-generated implicit social information.

In order to find out why using explicit social connections is more effective, we conduct the consistence analysis on
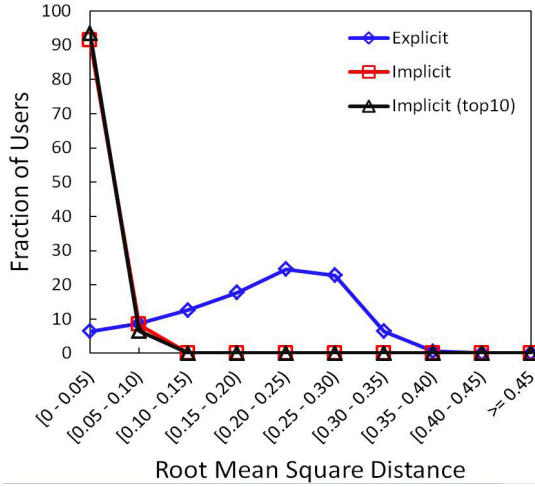
Figure 4: Similarity Consistence Analysis

three social networks we have, including one explicit (used in SR_{exp} method), and two implicit social networks (used in SR_{imp} and SR_{imp}^{top10} methods). The questions we address in this analysis are: How consistent are one user's social peers? Do the similarities between a user and his/her social peers vary a lot?

In order to answer the above questions, we evaluate the consistences based on the following metric, i.e., Root Mean Square Distance (RMSD). The definitions of user i are:

$$RMSD = \sqrt{\frac{\sum_{f \in S(i)}(s_{if} - \overline{s}_i)^2}{|S(i)|}}, \qquad (15)$$

where s_{ik} is the similarity between user i and user k defined in Equation 6, $\overline{s}_i$ is the average social similarity of user i, while $S(i)$ represents the list of social peers of user i.

From the definitions, we can see that we are actually measuring in what extent a user's social similarity s_{if} will deviate from his/her average social similarity $\overline{s}_i$. If a user's social peer similarities all fall into a small range, then his/her RMSD will be relatively small, which indicates this user's social peers are very consistent with this user. If we observe a large RMSD value, then this user's social peers are relatively diverse. Figure 4 shows the analysis results of RMSD.

We notice that the curves of these three social networks illustrate different patterns. The figure reveals that a very large portion of users in the generated two implicit networks have very small RMSD values, which implies that users' social peers are relatively more consistent in these two implicit social networks. The RMSD values in Douban explicit friend communities are relatively larger, which presents that users' social peers in this network are more diverse.

Combining the results we obtain from Table 7, this observation actually suggests that social network with larger friend diversity is more effective in improving recommendation quality. This conclusion also coincides with the intuition we propose in Section 4.3, i.e., dissimilar users can also be utilized to improve recommender systems.

6. CONCLUSION

This paper studies a research problem on how to improve recommender systems using implicit social information. A very general matrix factorization framework is employed to incorporate different implicit social information. The experimental analysis suggests that similar user information, dissimilar user information, similar item information and dissimilar item information can be effectively used to improve recommender systems. Our work not only provides in-depth insights to social recommendation techniques, but also will greatly extend the impact of previous and upcoming social recommendation approaches.

7. REFERENCES

[1] P. Bedi, H. Kaur, and S. Marwaha. Trust based recommender system for the semantic web. In *Proceedings of the 20th international joint conference on Artifical intelligence*, IJCAI'07, pages 2677–2682, Hyderabad, India, 2007.

[2] J. S. Breese, D. Heckerman, and C. Kadie. Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*, UAI'98, pages 43–52, Madison, Wisconsin, 1998.

[3] M. Deshpande and G. Karypis. Item-based top-n recommendation. *ACM Transactions on Information Systems*, 22(1):143–177, 2004.

[4] J. L. Herlocker, J. A. Konstan, A. Borchers, and J. Riedl. An algorithmic framework for performing collaborative filtering. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '99, pages 230–237, Berkeley, California, USA, 1999.

[5] T. Hofmann. Collaborative filtering via gaussian probabilistic latent semantic analysis. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, SIGIR '03, pages 259–266, Toronto, Canada, 2003.

[6] T. Hofmann. Latent semantic models for collaborative filtering. *ACM Transactions on Information Systems*, 22(1):89–115, 2004.

[7] J. Huang, X.-Q. Cheng, J. Guo, H.-W. Shen, and K. Yang. Social recommendation with interpersonal influence. In *Proceedings of the 19th European Conference on Artificial Intelligence*, ECAI '10, pages 601–606, Amsterdam, The Netherlands, 2010.

[8] M. Jamali and M. Ester. Trustwalker: a random walk model for combining trust-based and item-based recommendation. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '09, pages 397–406, Paris, France, 2009.

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

[10] R. Jin, J. Y. Chai, and L. Si. An automatic weighting scheme for collaborative filtering. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '04, pages 337–344, Sheffield, United Kingdom, 2004.

[11] A. Kohrs and B. Merialdo. Clustering for collaborative filtering applications. In *Proceedings of CIMCA '99*, 1999.

[12] Y. Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '08, pages 426–434, Las Vegas, Nevada, USA, 2008.

[13] Y. Koren. Collaborative filtering with temporal dynamics. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '09, pages 447–456, Paris, France, 2009.

[14] Y. Koren, R. M. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *IEEE Computer*, 42(8):30–37, 2009.

[15] M. Kurucz, A. A. Benczur, and K. Csalogany. Methods for large scale svd with missing values. In *Proceeding of KDD CUP '07*, San Jose, CA, USA, 2007.

[16] G. Linden, B. Smith, and J. York. Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet Computing*, pages 76–80, Jan/Feb 2003.

[17] N. N. Liu and Q. Yang. Eigenrank: a ranking-oriented approach to collaborative filtering. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '08, pages 83–90, Singapore, Singapore, 2008.

[18] H. Ma, I. King, and M. R. Lyu. Learning to recommend with explicit and implicit social relations. *ACM Transactions on Intelligent Systems and Technology*, 2(3):29:1–29:19, May 2011.

[19] H. Ma, M. R. Lyu, and I. King. Learning to recommend with trust and distrust relationships. In *Proceedings of the third ACM conference on Recommender systems*, RecSys '09, pages 189–196, New York, New York, USA, 2009.

[20] H. Ma, D. Zhou, C. Liu, M. R. Lyu, and I. King. Recommender systems with social regularization. In *Proceedings of the fourth ACM international conference on Web search and data mining*, WSDM '11, pages 287–296, Hong Kong, China, 2011.

[21] P. Massa and P. Avesani. Trust-aware collaborative filtering for recommender systems. In *Proceedings of CoopIS/DOA/ODBASE '04*, pages 492–508, 2004.

[22] P. Massa and P. Avesani. Trust-aware recommender systems. In *Proceedings of the 2007 ACM conference on Recommender systems*, RecSys '07, pages 17–24, Minneapolis, MN, USA, 2007.

[23] J. O'Donovan and B. Smyth. Trust in recommender systems. In *Proceedings of the 10th international conference on Intelligent user interfaces*, IUI '05, pages 167–174, San Diego, California, USA, 2005.

[24] J. D. M. Rennie and N. Srebro. Fast maximum margin matrix factorization for collaborative prediction. In *Proceedings of the 22nd International Conference on Machine Learning*, pages 713–719, Bonn, Germany, 2005.

[25] R. Salakhutdinov and A. Mnih. Probabilistic matrix factorization. In *Proceedings of Advances in Neural Information Processing Systems*, NIPS '07, 2007.

[26] R. Salakhutdinov and A. Mnih. Bayesian probabilistic matrix factorization using markov chain monte carlo. In *Proceedings of the 25th international conference on Machine learning*, ICML '08, pages 880–887, Helsinki, Finland, 2008.

[27] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*, WWW '01, pages 285–295, Hong Kong, Hong Kong, 2001.

[28] L. Si and R. Jin. Flexible mixture model for collaborative filtering. In *Proceedings of the 20th International Conference on Machine Learning*, ICML '03, pages 704–711, 2003.

[29] N. Srebro and T. Jaakkola. Weighted low-rank approximations. In *Proceedings of the 20th International Conference on Machine Learning*, pages 720–727, Washington, DC, USA, 2003.

[30] N. Srebro, J. D. M. Rennie, and T. Jaakkola. Maximum-margin matrix factorization. In *Proceedings of Advances in Neural Information Processing Systems*, NIPS '04, 2004.

[31] K. Yu, S. Zhu, J. Lafferty, and Y. Gong. Fast nonparametric matrix factorization for large-scale collaborative filtering. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '09, pages 211–218, Boston, MA, USA, 2009.

[32] Q. Yuan, L. Chen, and S. Zhao. Factorization vs. regularization: fusing heterogeneous social relationships in top-n recommendation. In *Proceedings of the fifth ACM conference on Recommender systems*, RecSys '11, pages 245–252, Chicago, Illinois, USA, 2011.

[33] Y. Zhang and J. Koren. Efficient bayesian hierarchical user modeling for recommendation system. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '07, pages 47–54, Amsterdam, The Netherlands, 2007.