

Two Models Based on Social Relations and SVD++ Method for Recommendation System

<https://doi.org/10.3991/ijim.v15i01.17751>

Ali M. Ahmed Al-Sabaawi (✉)
Aksaray University, Aksaray, Turkey
alisabaawi5@gmail.com

Hacer Karacan
Gazi University, Ankara, Turkey

Yusuf E. Yenice
Aksaray University, Aksaray, Turkey

Abstract—Recently, Recommender Systems (RSs) have attracted many researchers whose goal is to improve the performance of the prediction accuracy of recommendation systems by alleviating RSs drawbacks. The most common limitations are sparsity and the cold-start user problems. This article proposes two models to mitigate the effects of these limitations. The proposed models exploit five sources of information: rating information, which involves two sources, namely explicit and implicit, which can be extracted via users' ratings, and two types of social relations: explicit and implicit relations, the last source is confidence values that are included in the first model only. The whole sources are combined into the Singular Value Decomposition plus (SVD++) method. First, to extract implicit relations, each non-friend pair of users, the Multi-Steps Resource Allocation (MSRA) method is adopted to compute the probability of being friends. If the probability has accepted value which exceeds a threshold, an implicit relationship will be created. Second, the similarity of explicit and implicit social relationships for each pair of users is computed. Regarding the first model, a confidence value between each pair of users is computed by dividing the number of common items by the total number of items which have also rated by the first user of this pair. The confidence values are combined with the similarity values to produce the weight factor. Furthermore, the weight factor, explicit, and implicit feedback information are integrated into the SVD++ method to compute the missing prediction values. Additionally, three standard datasets are utilized in this study, namely Last. Fm, Ciao, and FilmTrust, to evaluate our models. The experimental results have revealed that the proposed models outperformed state-of-the-art approaches in terms of accuracy.

Keywords—Recommendation system, SVD++, social relations, data sparsity, cold-start

1 Introduction

Recommendation Systems (RSs) play a critical role in providing interested users' items among the mass of information that is available in online applications. Technically, a recommender system can be categorized in various fields, such as machine learning, text classification, and information retrieval [1][2]. The main aim of the RS is to find the relevant items by understanding users' behavior and predicting items that might be of interest to the user [3]. Recently, Collaborative Filtering (CF) has become one of the most common prototypes and most widely successful technologies in RS [4]. CF involves two types: memory-based and model-based. User-based and item-based are two techniques of the memory-based method, and both compute the similarity among users/items [5]. On the other hand, the model-based approach utilizes machine learning techniques to analyze users' activity and behavior in a particular online application, the system predicts items for that user according to his behavior [1]. However, CF faces a set of challenges, such as data sparsity, cold-start, and scalability. As data arise, which reaches thousands of users and items that remain without ratings, such a phenomenon is called a data sparsity problem. Moreover, the cold-start problem is the issue of the lack of ability to make predictions for new items or users. Furthermore, there is no history information on cold-starts of users/items; thus, RSs cannot handle this problem [5]. To curtail such shortcomings, many data mining methods have been adopted, including clustering CF and matrix factorization (MF) [6], MF can find the best latent factor in the user-item matrix. MF includes many methods, such as Singular Value Decomposition SVD [7], Probabilistic Matrix Factorization (PMF) [8] and [9], Non-negative Matrix Factorization (NMF) [10], and SVD++ [1]. Many studies have been made to overcome further the problems of RSs. These studies exploited different techniques and integrated various sources such as social relations, context, and personal information to enhance the accuracy of the RS. Social relations have widely been used recently in the RS field [11] and [12], in which the user can return to his/her friends' relationships before deciding to accept or reject a particular item. Several social factors were applied, such as reliability-based Trust-Aware Collaborative Filtering (TACF) [13], explicit relationships Social Recommendation [14], explicit implicit relationships using Probabilistic Matrix Factorization (EISR) [9], and MSRA-SVD [5]. Explicit relations are the main source of social information. Here, a user can find his interesting items by exploiting the preferences items of his/her explicit friends. Despite the explicit relations contributing to alleviate the data sparsity problem, the problem still remains because most of the users have a limited number of relations. Accordingly, another factor can be adopted which has massive information called implicit relations [15].

In real life, users have similar preferences to each other even when there are no direct relations between them in the network. Many methods have utilized explicit relations and others have exploited implicit relations to overcome the data sparsity problem, but still require further sufficient prediction accuracy. It is poor practice to depend only on direct connections in the social network, as proposed by [14], and indirect relations with one step of the users' neighbourhood only [9]. The limitation of [9] is neglecting information of other steps of the indirect connection which may impli-

cate vital information. In addition, social interactions among users are established based on trust relations which can be exploited as extra factors to reduce the sparsity problem [13]. Moreover, the limitations of [5] ignored the implicit feedback information, and the trust relations of the users were not exploited.

In this study, the foremost goal increases the accuracy by overcoming the previous limitations, two models are proposed. Both methods adopted four sources of information: implicit relations which are identified by Multi-Steps Resource Allocation (MSRA), explicit social relations, explicit feedback, and implicit feedback, which can be extracted via rating information. These sources are integrated into the SVD++ method to create the MSRA-SVD++ model. Finally, the confidence factor between users, which is the fifth source of information, it is utilized in the first proposed model only, which is applied and combined with the similarity values to produce the weight-SVD++ model.

2 Material and Methods

In this section, the methods and sources that are used in this research are explained. SVD++ method is demonstrated in the next subsection. Whereas the link prediction technique is explained in 2.2. Social relations are clarified in subsection 2.3. Finally, the two proposed models are highlighted in subsections 2.4 and 2.5.

2.1 SVD++ method

The SVD++ method, which is utilized by many researchers, is derived from SVD. However, the SVD method exploits explicit information (votes or ratings) that are available in user-item matrices and ignores the implicit information such as clicking, browsing, and purchasing [1]. SVD++ proposed by [16]. For example, assume a 2D user-item matrix (M), wherein the rows p indicates the users, and the columns q represents the items. The rating value where the users can rate the items can be expressed as $r_{u,v}$. The purpose of the RS is to predict items for users that might interest them. In general, the dataset is huge and many users leave the items without a rating, thus the RS faces a data sparsity problem. However, SVD++ decomposes the user-item matrix into two matrices, p and q , which are orthogonal matrices to represent the users and items, respectively. The decomposition process converts the high dimensional user-item matrix that contains all the data into low-dimensional data $p_{u,f}$ and $q_{i,f}$, where u is the user number, i the item number, and f indicates the dimension after reduction. In other words, SVD++ reduces the size of matrices p and q to a specific value and finds the best low-dimensional data representation of the user-item matrix by removing any unimportant data. In addition, SVD++ is an expanded version of SVD. The difference is that the implicit feedback is taken into consideration in SVD++, whereas the SVD method depends only on users' ratings (explicit feedback). Subsequently, a binary matrix is constructed to represent the implicit feedback, where the new matrix is separated into left and right orthogonal matrices. The following formula shows the prediction of SVD++:

$$r_{u,i} = \mu + b_u + b_i + q^T \left(p_u + \frac{1}{\sqrt{|N_u|}} \sum_{j \in N_u} y_j \right) \quad (1)$$

$$b_i = \frac{(\sum_{u \in R(i)} r_{u,i} - \mu)}{|R_i|} \quad (2)$$

$$b_u = \frac{(\sum_{u \in R(u)} r_{u,i} - \mu - b_i)}{|R_u|} \quad (3)$$

where μ is the average value of the whole dataset, b_u and b_i are the observed deviations from the mean value of user u and item i , respectively. $|N_u|$ is the number of items rated by user u , $|R_u|$ is the number of users that rated a specific item, $|R_i|$ is the number of items that were rated by a set of users, and y_j is the left orthogonal matrix of the implicit matrix. [17] proposed two regularization parameters (λ_1 and λ_2) which can be added to $|R_u|$ and $|R_i|$ to compute the user bias and item bias. Eqs (2) and (3), are adopted in this study and modified thus:

$$b_i = \frac{(\sum_{u \in R(i)} r_{u,i} - \mu)}{\lambda_1 + |R_i|} \quad (4)$$

$$b_u = \frac{(\sum_{u \in R(u)} r_{u,i} - \mu - b_i)}{\lambda_2 + |R_u|} \quad (5)$$

2.2 Link prediction

Social networks encompass a large amount of information, including explicit information such as direct relations (trust relations) and indirect relations (friend's relations). Another kind of information is implicit relations. Regarding the implicit relations, the link prediction technique is exploited to predict missing information in a social network. Many methods are utilized to find the implicit information by predicting new links in a social network, such as Resource Allocation Index (RAI) [15] and Multi-Steps Resource Allocation (MSRA) [18]. According to [5], MSRA achieved better results than RAI, CN and JC such that it can predict implicit relations with an acceptance ratio of the accuracy (for more details, review reference [5]). MSRA accommodates a multi-step process, but in this work two-steps resource transmission is employed to achieve the implicit relations, thus:

$$s_{xy}^{MSRA2} = s_{xy}^{MSRA1} + \sum_{z1 \in Z1} \sum_{z2 \in Z2} \frac{1}{k_{z1}} \frac{1}{k_{z2}} \quad (6)$$

$$s_{xy}^{MSRA1} = \sum_{z \in |r(x) \cap r(y)|} \frac{1}{|r(z)|} \quad (7)$$

$r(x)$ and $r(y)$ indicate the sets of the friends of x and y , respectively. $Z1$ refers to $r(x) \cap r(y)$ and $Z2$ to $r(z1) \cap r(y)/Z1$.

2.3 Social relations

Social relations provide solutions to overcome the data sparsity problem in RS. In reality, users prefer to ask their friends before deciding on a certain action. The same aspect can be employed in RS. Recently, many applications allow how users to establish their friends' relationships and the RS can exploit these relations to enhance the performance. Several studies have utilized social relations to boost prediction accuracy. [14] utilized explicit social relations. For each pair of users, the similarity technique was applied to calculate the similarity between them. Finally, the model was combined with MF. Similarly, in EISR was proposed by [9], explicit and implicit relations were exploited. The same scenario of the previous study was followed, but here implicit relations were added as an extra factor. In [5], the implicit and explicit relations were used, but the difference is using MSRA algorithm instead of RAI algorithm, then the similarity was computed and all the social factors with rating information were incorporated into SVD method. Subsequently, the last two studies were based on implicit relations to represent implicit friends' relationships, and these relations concluded by link prediction technique that finds missing relations in a social network. Another kind of social relations is confidence statements, which can be divided into explicit trust and implicit trust statements [19]. Unlike friend relations, a trust statement means directed relations, such as when user x trusts user y , it is not necessary for user y trusts user x . In some datasets, an explicit trust may be unavailable; therefore, the implicit trust can be estimated from a user-item matrix, by monitoring the attitude of the users through their ratings that lead to conclude new resource information, which may increase the prediction accuracy. The following formula proposed by [20] shows the calculation of implicit trust.

$$T_{u,v} = A_{u,v}/A_u \quad (8)$$

Where A_u is the number of items that are rated by user u . $A_{u,v}$ is the set of common items rated by u and v .

According to [13], the trust statement can be integrated into the rated information to determine the sparsity problem. Thus, the adjusted weight between users u and v is computed thus:

$$w_{u,v} = \begin{cases} \frac{2 \cdot \text{sim}_{u,v} \cdot T_{u,v}}{\text{sim}_{u,v} + T_{u,v}}, & \text{sim}_{u,v} \cdot T_{u,v} \neq 0, \\ & \text{and } \text{sim}_{u,v} + T_{u,v} \neq 0 \\ \text{sim}_{u,v}, & \text{sim}_{u,v} \neq 0, \text{ and } T_{u,v} = 0 \\ T_{u,v}, & T_{u,v} \neq 0, \text{ and } \text{sim}_{u,v} = 0 \\ 0, & \text{otherwise} \end{cases} \quad (9)$$

Where the $\text{sim}_{u,v}$ denotes the similarity between users u and v , which computes using the Pearson Correlation Coefficient (PCC) as the following.

$$\text{sim}(u, v) = \frac{\sum_{h=1}^n (r_{u,jh} - \bar{r}_u)(r_{v,jh} - \bar{r}_v)}{\sqrt{\sum_{h=1}^n (r_{u,jh} - \bar{r}_u)^2} \sqrt{\sum_{h=1}^n (r_{v,jh} - \bar{r}_v)^2}} \quad (10)$$

Here, $r_{x,y}$ is the rating value of item j that is rated by user u . \bar{r}_x indicates the mean rating of the user x , and n refers to the items that are rated by both users u and v . The similarity value between u and v ranges over $[-1,1]$, so a higher value means high similarity. Although the preceding methods improved the RS performance, it still lacks adequate accuracy. The following subsections propose how to overcome this limitation.

2.4 Weight-SVD++ model

Five sources of information are used to feed the matrix factorization technique, namely the rating file that includes explicit and implicit feedback information, explicit relations, implicit relations, and confidence values. The rating information and explicit social relations sources are obtainable directly from the rating file and social file respectively, while the implicit relations and confidence values sources are computed. Regarding the implicit relations, MSRA is adapted to discover the implicit relations of users within the social network. The user not only can be affected by his friends' taste but also by other users, even if there is a distance between him and them in the network, and especially when those users are friends of his/her friends. Therefore, MSRA is utilized to apply a link prediction technique and predict missing relations that may contain vital information.

In addition, confidence values can be exploited to reduce the sparsity problem and consequently increase the prediction accuracy. After acquiring the implicit relations via MSRA, a binary user-user matrix is created, one's values refer to explicit/implicit relations. Afterward, the similarity of the explicit relations and implicit relations is computed using the Pearson correlation coefficient (PCC) which is uses rating values of the users, as shown in Eq. (10). The output after applying the similarity is another user-user matrix with the computed similarity values.

Furthermore, the confidence value is computed among all users using Eq. (8); a user-user matrix is created that has confidence values. Later, the weight values are computed by integrating the user-user matrix of the similarity values, and the user-user matrix of the confidence values, by Eq. (9). Eventually, the weight values and rating values are integrated into the SVD++ method to produce the weight-SVD++ model, details of which are shown in Figure 1. The optimization technique is adopted using Stochastic Gradient Descent (SGD) method, which is used to regularize the square error as follows:

$$\frac{\partial E}{\partial u_i} = \sum_{(u,i) \in K} \left(r_{u,i} - \mu - b_i - b_u - q_i^t \left(p_u + \frac{1}{\sqrt{|N_u|}} \sum_{j \in N_u} y_j \right) \right)^2 + \lambda (b_u^2 + b_i^2 + \|p_u\|^2 + \|q_i\|^2 + \|y_j\|^2) + \beta_w \sum_{i=1}^u \sum_{f \in F(i)} W(i, f) \|p_i - p_f\|_{fro}^2 \quad (11)$$

$$\frac{\partial E}{\partial v_i} = \sum_{(u,i) \in m} \left(r_{u,i} - \mu - b_u - b_i - q_i^t \left(p_u + \frac{1}{\sqrt{|N_u|}} \sum_{j \in N_u} y_j \right) \right)^2 + \lambda (b_i^2 + b_u^2 + \|p_u\|^2 + \|q_i\|^2 + \|y_j\|^2) \quad (12)$$

Where β_w is a constant value that is used to determine the influence degrees of the weight values. $F(i)$ denotes the numbers of users that have weights of user i ; and $W(i,f)$ refers to the weight value between users i and f . λ is the constant value that determines the degree of the constraint. These equations are utilized to find the best value for vectors p_u , q_u , b_u , b_i and y_i vectors. The following equations show the updating values after each loop:

$$p_u = p_u + \gamma \left(\frac{\partial E}{\partial p_u} q_i - \lambda p_u \right) \quad (13)$$

$$q_i = q_i + \gamma \left(\frac{\partial E}{\partial q_i} \left(p_u + \frac{1}{\sqrt{|N_u|}} \sum_{j \in N_u} y_j \right) - \lambda q_i \right) \quad (14)$$

$$y_j = y_j + \gamma \left(\frac{\partial E}{\partial y_j} \left(\frac{q_i}{\sqrt{|N_u|}} \right) - \lambda y_j \right) \quad (15)$$

$$b_u = b_u + \gamma \left(\frac{\partial E}{\partial b_u} - \lambda b_u \right) \quad (16)$$

$$b_i = b_i + \gamma \left(\frac{\partial E}{\partial b_i} - \lambda b_i \right) \quad (17)$$

Here, γ is the learning rate. Then, the prediction value is computed according to Eq. 1. Algorithm 1 shows the implementation steps of the weight-SVD++ model.

Algorithm 1: Weight-SVD++ model:

1. Read the rating file and create a user-item matrix.
2. Read the social file and create a user-user matrix as explicit relations.
3. Apply the MSRA algorithm using Eqs. (6) and (7) and create a user-user matrix of implicit relations.
4. Compute the similarity between each pair of users who have an explicit relation or an implicit relation using Eq. (10).
5. Find the confidence values of every pair of users using Eq. (8).
6. Calculate the weight values, which combines the similarity values and confidence values using Eq. (9).
7. Initialize random latent factor matrices for vectors p and q .
8. Compute b_i and b_u vectors using Eqs. (4) and (5).
9. Compute the regularization square error using Eqs. (11) and (12).
10. Update p_u , q_i , y_i , b_u , b_i by using Eqs. (13), (14), (15), (16), (17) respectively.
11. Compute the prediction values using Eq. (1).

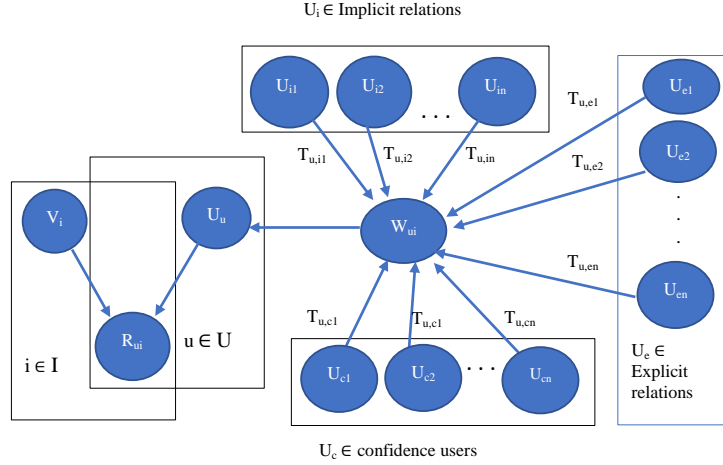


Fig. 1. Proposed weight SVD++ model considering explicit and implicit relations and confidence values

2.5 MSRA-SVD++ model

The second model involves four sources of information, namely explicit feedback, implicit feedback, explicit relations, and implicit relations. Explicit and implicit feedback information can be derived from the rating matrix, whereas the explicit relations are available directly in the social file. Implicit relations can be computed using the MSRA algorithm. The difference between the first model and the current model is the utilization of the confidence values. In this model, the similarity values of the implicit and explicit relations are adopted only. In general, the same equations, which are used in the previous model, are applied here with a simple difference in Eq. (11), which is modified as follows:

$$\begin{aligned} \frac{\partial E}{\partial U_u} = & \sum_{(u,i) \in K} \left(r_{u,i} - \mu - b_i - b_u - q_i^t \left(p_u + \frac{1}{\sqrt{|N_{iu}|}} \sum_{j \in N_{iu}} y_j \right) \right)^2 + \lambda (b_u^2 + b_i^2 + \\ & \|p_u\|^2 + \|q_i\|^2 + \|y_j\|^2) + \beta_e \sum_{i=1}^u \sum_{f \in F(i)} \text{sim}(i, f) \|p_i - p_f\|_{fro}^2 + \\ & \beta_i \sum_{i=1}^u \sum_{f^* \in F^*(i)} \text{sim}(i, f^*) \|p_i - p_{f^*}\|_{fro}^2 \end{aligned} \quad (18)$$

where β_e and β_i are constant values greater than zero that used to control the impact of the explicit and implicit relationships, respectively, $F(i)$ denotes the numbers of explicit friends and $F^*(i)$ indicates implicit relationships of user i ; and $\text{sim}(x,y)$ is the similarity value between users x and y . λ is the constant value that determines the degree of the constraint.

Algorithm 2: MSRA-SVD++ model:

1. Read the rating file and create a user-item matrix
2. Read the social file and create a user-user matrix as explicit relations

3. Apply the MSRA algorithm using Eqs. (6) and (7), then create a user-user matrix of implicit relations
4. Compute the similarity between each pair of users who have an explicit relation or an implicit relation using Eq. (10)
5. Initialize random latent factor matrices for vectors p and q
6. Compute b_i and b_u vectors using Eqs. (4) and (5)
7. Compute the regularization square error using Eqs. (18) and (12)
8. Update p_u, q_i, y_j, b_u, b_i by using Eqs. (13), (14), (15), (16), (17) respectively.
9. Compute the prediction values using Eq. (1)

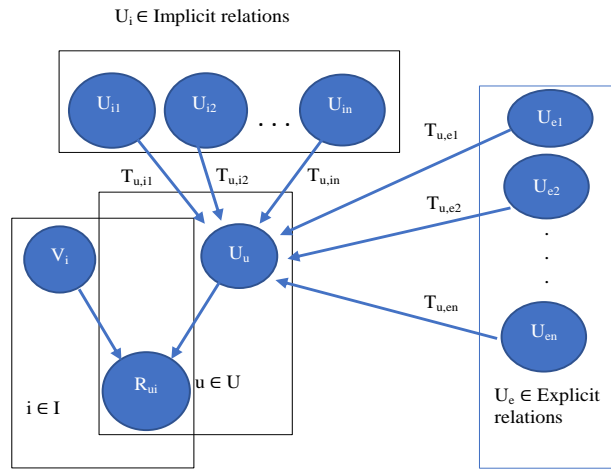


Fig. 2. Proposed MSRA-SVD++ model considering explicit and implicit relations

3 Results and Discussion

The two detailed implementations of the two models are presented in this section, including the dataset description, evaluation metrics, parameter settings, the benchmarking of the experiments, and the impact of using social relations.

3.1 Dataset description

In this study, three standard datasets are used: Last. Fm, Ciao, and FilmTrust. Last. Fm is an online music system that enables users to create their relationships. This dataset was released for the first time in 2011 at the HetRec workshop [21], Last.FM has 2100 users and 18745 items. In this study, users with greater than four relationships are included. Thus, the number of users is 1123 and 17892 items. Last. Fm does not have rating values but instead, it has listening counts number and Eq. (19), proposed by [22], is applied to map these values from 1 to 5 to present the rating values.

$$r = \begin{cases} \lfloor \log_{10} l \rfloor + 1, & \text{if } \lfloor \log_{10} l \rfloor + 1 \leq 5 \\ 5, & \text{otherwise} \end{cases} \quad (19)$$

Here, r is the rating integer value, l denotes the listening count, and $\lfloor x \rfloor$ denotes the operation of rounding towards zero.

The second dataset is Ciao, which was gathered by [23] and involves of 7375 users and 99746 items. The rating value includes integer values 1-5. A subset of the Ciao dataset is utilized in this study to avoid memory consuming in which the items with ratings of less than two are ignored and users that have one or more ratings with at least one relationship are selected. Therefore, the size of the Ciao dataset became 6767 users and 22229 items. Finally, FilmTrust is the third dataset to be applied. It is a movie rating and trust social relations dataset. FilmTrust was produced by [24]. The ratings are float values [0.5 – 4] with 0.5 increments. Table 1 depicts the statistical of these datasets.

Table 1. Statistics of the Ciao, FilmTrust, and Last. Fm datasets

Dataset	Ciao	FilmTrust	Last.FM
# users	6767	1508	1123
# items	22229	2071	18745
# rating	185759	35497	55,140
% Density	0.0012	1.14	0.0026
# explicit relation	111780	1853	11064
#implicit relation using MSRA	610710	3138	177987

3.2 Evaluation metrics

Two well-known metrics are utilized to evaluate the performance of the proposed method, namely the Mean Absolute Error (MAE) and the Root Mean Square Error (RMSE), which are defined by the following formulas:

$$MAE = \frac{1}{N} \sum_{i=1}^N |r_{u,i} - \bar{r}_{u,i}| \quad (20)$$

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (r_{u,i} - \bar{r}_{u,i})^2} \quad (21)$$

where N is the number of the whole prediction, $r_{u,i}$ is the real rated value in the user-item matrix for user u and item i , and $\bar{r}_{u,i}$ is the computed value of the proposed models. High accuracy results mean the values of both metrics are close to zero.

3.3 Experimental setup

The determined optimal experimental settings in this study are shown in the following Table 2.

Table 2. Parameters setup

Dataset	Ciao	FilmTrust	Last.FM
β_e	0.001	0.005	0.002
β_i	0.001	0.005	0.002
β_w	0.002	0.002	0.002
α	0.006	0.015	0.01
λ_e and λ_i	0.0001	0.001	0.001
λ_1 and λ_2	4, 8	7, 10	10, 25
d	5, 10	5, 10	5, 10
No. of Iterations	5	5	5

To evaluate the proposed methods, state-of-the-art approaches are benchmarked against our study, including RSTE [25], PMF [26], SVD++ [27], SocReg [14], TrustMF [28], MSRA-SVD [5], and UE-SVD++ [29].

In the literature, the most similar studies of this study are TrustMF, SocReg, MSRA-SVD, and UE-SVD++. Thus, for equitable comparison, the proposed models were tested under the same conditions, such as the number of iterations, explicit social relations, the dimensional reduction value, and the number of users. Table 2 shows the parameters setup. Two testing views are displayed in this study. The first view includes all users who have at least one relationship (for Ciao and Last.FM) and one rating (for all datasets), the users were distributed into training and testing parts randomly. The second view is the Cold-start users, which means users with ratings of less than five were included in the test part for the Ciao and FilmTrust datasets, whereas only three ratings of 20% of all users in the Last. Fm dataset are randomly selected.

3.4 Benchmarking

5-fold Cross-validation was adopted in this experiment. The datasets were divided randomly into five folds in which four-folds had been used for training (80%) and one for testing (20%). To guarantee test all the data, each execution had been performed five times and the average was computed to produce the outcome of each implementation. Moreover, the experiment was implemented several times and the mean was produced to display the eventual result. Tables 3 and 4 depict the accuracy of the results for all users and cold-start users, respectively. The accuracy results were computed through the MAE and RMSE metrics. For both metrics, high values indicated low accuracy. On the other hand, if the values had been close to zero, the results would have been highly accurate. Two results were displayed in both tables when the number of dimensions (d) is 5 and 10. The proposed models were compared with state-of-the-art studies, with PMF, SVD++, and UE-SVD++ that used only rated information and the other studies that utilized social relations. All these studies are shown in Tables 3 and 4. Boldface implies the best results and (*) indicates the second-best results.

Tables 3 and 4 show the influences of social relations to improve the prediction process. Accordingly, comparing the results of PMF and SVD++ against TrustMF,

SocReg, and MSRA-SVD models depict how social relations have a vital rule to boost accuracy. Moreover, the effect of the social relations in cold-start users (Table 4) is greater than that in all users (Table 3). To clarify, cold start users have insufficient rating history to utilize it in the prediction process, whereas for all users, there is reasonable rating history that can be exploited to predict new items for them, and using social relations assist the prediction, but to smaller degree.

According to Table 3, the proposed models outperformed all the previous approaches for both models (weight-SVD++ and MSRA-SVD++) and the UE-SVD++ has the closest results in FilmTrust dataset. However, MSRA-SVD has the nearest results to our proposed models in Ciao and Last. Fm datasets. Moreover, MSRA-SVD++ achieved the best performance such that it outperformed our first model (weight-SVD++) in the FilmTrust dataset. Although the weight-SVD++ model uses many sources of information to reduce the sparsity, the accuracy results were lower than the MSRA-SVD++ results. To justify the reason, the weight-SVD++ model uses confidence values as additional information sources to find relations between users. Additionally, dimension reduction methods such as SVD++ achieve good performance when datasets have acceptable sparsity ratios. In contrast, when the sparsity is reduced to a minimum level (as performed by weight-SVD++), the results will be affected negatively especially with high-density datasets, such as FilmTrust. Nonetheless, the weight SVD++ model nearly outperformed the MSRA-SVD++ model in terms of MAE and RMSE in the Ciao and Last.FM datasets because these datasets have higher sparsity than FilmTrust, so reducing the sparsity using social relations and confidence values achieved higher accuracy. For example, the improvement that is achieved by the MSRA-SVD++ model comparing with the second-best results in Ciao is about 0.63% and 1.11% for MAE and RMSE respectively when $d = 5$, and it is about 0.57% for MAE and 1% for RMSE when $d = 10$. Likewise, the improvement in FilmTrust is approximately 1.4% For MAE and 0.07% for RMSE when d equals 5, Also it is 1.24% and 0.32% for MAE and RMSE respectively when d equals 10.

Table 4 presents the results of cold-start users, who have less than five ratings. Table 4 reveals that MSRA-SVD++ once again exceeded the weight-SVD++ model in FilmTrust. It accomplished the best results in terms of MAE and RMSE, and the improvement that is achieved by comparing MSRA-SVD++ over the best-second results, when $d=10$, is about 3.03% and 3.38% for MAE and RMSE in FilmTrust, 3.29% for MAE and 3.83% for RMSE in FilmTrust, and in Last. Fm dataset the improvement of around 5.29% for MAE and 0.79% for RMSE. Although MSRA-SVD and our present studies are adopted MSRA algorithm to extract implicit relations, the present study has significant improvement in terms of accurate results. Two different reasons caused this improvement. First, adding λ_1 and λ_2 during the computation of user-bias and item-bias, respectively as shown in Eqs. (4) and (5). Second, the proposed method used implicit feedback information source as one of the five sources employed in this study, whereas, MSRA-SVD ignored this source.

Additionally, the weight-SVD++ model is superior to MSRA-SVD++ for Ciao and Last.FM datasets in terms of RMSE and MAE such that the improvement that is achieved by weight-SVD++ over MSRA-SVD++ is around 0.03% and 0.04% when d

= 5 in terms of MAE and RMSE, respectively in Ciao. The same superiority can be shown in the Last. Fm dataset, which is 0.57% for MAE and 0.14 for RMSE.

In summary of the above results, the weight-SVD++ model can achieve better performance when the dataset has less density such that the density is 0.0012% and 0.0026% for Ciao and Last.FM respectively. Furthermore, with high-density datasets such as FilmTrust, which is 1.14%, extra sources of information have an adverse impact; thus, the MSRA-SVD++ model produces higher accuracy results than the weight-SVD++ model. Subsequently, for the dataset with high sparsity, the cold-start user problem can be alleviated using social relations with confidence values, which indicates a vital source of information and it has more effectiveness than the dataset with a lower sparsity ratio. Accordingly, the proposed methods enhanced the results in terms of accuracy and overcame the data sparsity and cold-start user problems by exploiting four/five sources of information: explicit social relations, implicit social relations, explicit rating values, implicit rating feedback (for MSRA-SVD++ model), with confidence values (for weight-SVD++ model). All the results confirm that the proposed method succeeded to enhance the recommendation system performance.

Table 3. Evaluation performance for all users

Methods	Metrics	Ciao		FilmTrust		Last FM	
		d=5	d=10	d=5	d=10	d=5	d=10
PMF	MAE	0.920	1.078	0.714	0.735	0.427	0.426
	RMSE	1.206	0.822	0.949	0.968	0.537	0.535
SVD++	MAE	0.752	0.748	0.613*	0.611*	0.413	0.412
	RMSE	1.013	1.001	0.804	0.802*	0.533	0.532
RSTE	MAE	0.767	0.763	0.628	0.640	-----	-----
	RMSE	1.020	1.013	0.810	0.835	-----	-----
SocReg	MAE	0.899	0.815	0.674	0.668	0.422	0.422
	RMSE	1.183	1.076	0.878	0.875	0.531	0.530
Trust MF	MAE	0.742	0.753	0.631	0.631	-----	-----
	RMSE	0.983	1.014	0.810	0.819	-----	-----
MSRA-SVD	MAE	0.7274*	0.7263*	0.6214	0.6174	0.3944*	0.3931*
	RMSE	0.9589*	0.9572*	0.8351	0.8325	0.5182*	0.5177*
UE-SVD++	MAE	-----	-----	0.6203	0.6280	-----	-----
	RMSE	-----	-----	0.8026*	0.8120	-----	-----
Weight-SVD++	MAE	0.7222	0.7218	0.6052	0.6056	0.3403	0.340
	RMSE	0.9474	0.9467	0.8022	0.8017	0.458	0.4575
MSRA-SVD++	MAE	0.7228	0.7221	0.6044	0.6035	0.3405	0.3403
	RMSE	0.9482	0.9476	0.802	0.7994	0.4582	0.4579

Table 4. Evaluation performance for cold-start users

Methods	Metrics	Ciao		FilmTrust		Last FM	
		<i>d</i> =5	<i>d</i> =10	<i>d</i> =5	<i>d</i> =10	<i>d</i> =5	<i>d</i> =10
PMF	MAE	1.033	0.926	0.814	0.767	----	----
	RMSE	1.334	1.191	1.079	1.009	----	----
SVD++	MAE	0.759*	0.749*	0.677	0.680	0.4281	0.4272
	RMSE	1.039	1.020	0.897	0.905	0.5572	0.5560
RSTE	MAE	0.766	0.764	0.680	0.674*	0.4268	0.4251
	RMSE	1.101	1.074	0.884	0.900	0.5691	0.5672
SocReg	MAE	1.173	0.949	0.881	0.771	-----	-----
	RMSE	1.430	1.214	1.104	1.034	-----	-----
Trust MF	MAE	0.752	0.770	0.674*	0.687	----	----
	RMSE	0.954	1.096	0.867*	0.900	----	----
MSRA-SVD	MAE	0.7662	0.7658	0.6826	0.6821	0.4233*	0.4228*
	RMSE	0.988*	0.9873*	0.8967	0.8965*	0.5462*	0.5447*
Weight-SVD++	MAE	0.7267	0.7266	0.6628	0.6614	0.4003	0.3994
	RMSE	0.9493	0.9489	0.8739	0.8747	0.5417	0.5395
MSRA-SVD++	MAE	0.727	0.7263	0.6526	0.6518	0.4026	0.4004
	RMSE	0.9499	0.9494	0.8627	0.8621	0.5425	0.5404

3.5 Impact of social relations

Figures 3, 4, and 5 display the results of the RMSE metric of different beta values for the MSRA-SVD++ model when dimension value (*d*) is 10, for Ciao, Last. Fm, and FilmTrust datasets respectively. Beta values are utilized along with similarity values between users, more specifically beta values control the degree of using social relations to reduce the regularization error during the training phase. High beta values refer to using a high similarity ratio between users who have relationships. In this study, a set of beta values is used, which are (0.0001, 0.001/0.002, 0.01, 0.02, 0.03, 0.04, and 0.05) for Ciao and Last. Fm, and (0.0001, 0.005, 0.1, 0.2, 0.3, 0.4, and 0.5) for FilmTrust. As shown in Figures 3-5, the RMSE results are varied. But in general, high beta values cause low accuracy (high values of RMSE), also small beta values lead to low accuracy. The best beta values to produce the highest accuracy are different from one dataset to another based on the sparsity ratio and the level of similarity among users. For example, the best beta value, in general, is 0.001 for Figure 3, where the result is 0.9476. In contrast, the best beta value in Figure 4 is 0.002, the result is 0.4575. Likewise, the beta value is different in Figure 5, the best result is reported when beta equals 0.005, the result is 0.7995.

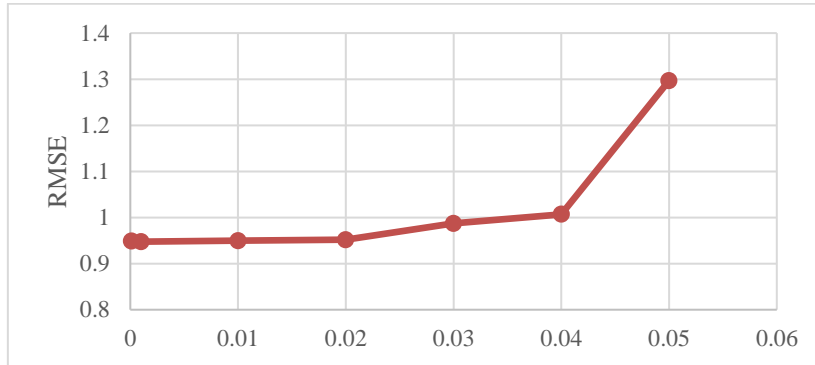


Fig. 3. Results of RMSE for different beta values on Ciao

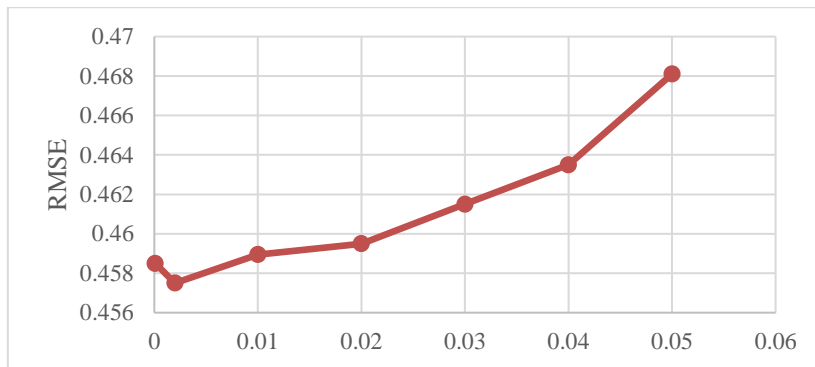


Fig. 4. Results of RMSE for different beta values on Last. Fm

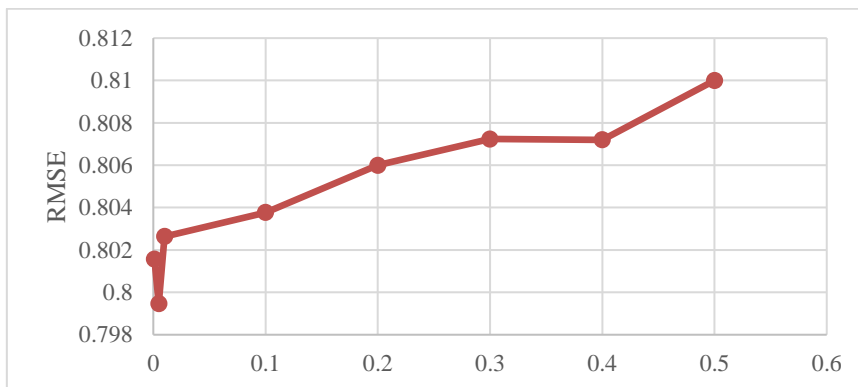


Fig. 5. Results of RMSE for different beta values on FilmTrust

4 Conclusion

This paper proposed two models by integrating into the first model: rating information, which indicates explicit and implicit ratings, social relations that involve explicit and implicit social relations, also confidence values that are computed by measuring the common rating items of each pair of users. Regarding the second model, the rating and social relations were exploited without confidence values. Both models were tested via three standard datasets, which were utilized to evaluate the performance of our approach. The results revealed that the dimension reduction method presents better performance when it works with an acceptable level of density than working with high density (preferring sparsity over density). Additionally, the comprehensive experimental outcome showed that social relations have further influence when it was applied to cold-start users, who needed extra information to find their preferences. Therefore, the impact of social relations for the cold-start users is more obvious than for those who have reasonable rating history in the same dataset. Moreover, the effectiveness of the social relations increases for the low-density ratio datasets, where social relations contribute to enriching the dataset with more information that can support the prediction process. For further development, the confidence equation in the weight-SVD++ model can be modified to keep the density ratio at an acceptable level, or applying another similarity equation. Eventually, the clustering model may improve accuracy by separating users into different communities.

5 References

- [1] J. Jiao, X. Zhang, F. Li, and Y. Wang, "A Novel Learning Rate Function and Its Application on the SVD++ Recommendation Algorithm," *IEEE Access*, vol. 8, pp. 14112–14122, 2019. <https://doi.org/10.1109/access.2019.2960523>
- [2] S. Sarif, Z. Hanafi, S. Farvin, S. Zaibon, and M. Mohsin, "Makan@Local Chatok: Mobile Eatery Recommendation System Based on Local Knowledge," *Int. J. Interact. Mob. Technol.*, vol. 14, p. 129, Apr. 2020, <https://doi.org/10.3991/ijim.v14i05.13357>
- [3] R. K. Dewi, M. T. Ananta, L. Fanani, K. C. Brata, and N. D. Priandani, "The development of mobile culinary recommendation system based on group decision support system," *Int. J. Interact. Mob. Technol.*, vol. 12, no. 3, pp. 209–216, 2018. <https://doi.org/10.3991/ijim.v12i3.7799>
- [4] X. Su and T. M. Khoshgoftaar, "A Survey of Collaborative Filtering Techniques," *Adv. Artif. Intell.*, vol. 2009, no. Section 3, pp. 1–19, 2009, doi: 10.1155/2009/421425.
- [5] A. M. A. Al-Sabaawi, H. Karacan, and Y. E. Yenice, "Exploiting implicit social relationships via dimension reduction to improve recommendation system performance," *PLoS One*, vol. 15, no. 4, p. e0231457, Apr. 2020, [Online]. Available: <https://doi.org/10.1371/journal.pone.0231457>.
- [6] J. Lin, X.-H. Yan, and B. Huang, "Collaborative filtering recommendation algorithm based on SVD and fuzzy clustering," *Comput. Syst. Appl.*, vol. 25, no. 11, pp. 156–163, 2016.
- [7] X. Zhou, J. He, G. Huang, and Y. Zhang, "SVD-based incremental approaches for recommender systems," *J. Comput. Syst. Sci.*, vol. 81, no. 4, pp. 717–733, 2015. <https://doi.org/10.1016/j.jcss.2014.11.016>

- [8] Y. Cao, W. Li, and D. Zheng, “An improved neighborhood-aware unified probabilistic matrix factorization recommendation,” *Wirel. Pers. Commun.*, vol. 102, no. 4, pp. 3121–3140, 2018. <https://doi.org/10.1007/s11277-018-5332-2>
- [9] W. Reafee, N. Salim, and A. Khan, “The power of implicit social relation in rating prediction of social recommender systems,” *PLoS One*, vol. 11, no. 5, 2016. <https://doi.org/10.1371/journal.pone.0154848>
- [10] . Li, C. Chen, H. Chen, and C. Tong, “Towards context-aware social recommendation via individual trust,” *Knowledge-Based Syst.*, vol. 127, pp. 58–66, 2017. <https://doi.org/10.1016/j.knosys.2017.02.032>
- [11] U. Liji, Y. Chai, and J. Chen, “Improved personalized recommendation based on user attributes clustering and score matrix filling,” *Comput. Stand. Interfaces*, vol. 57, pp. 59–67, 2018. <https://doi.org/10.1016/j.csi.2017.11.005>
- [12] S. Rautela and A. Sharma, “Spirituality and Social Media: Connecting the Dots,” *Int. J. Interact. Mob. Technol.*, vol. 13, p. 81, Sep. 2019, <https://doi.org/10.3991/ijim.v13i09.8863>
- [13] P. Moradi and S. Ahmadian, “A reliability-based recommendation method to improve trust-aware recommender systems,” *Expert Syst. Appl.*, vol. 42, no. 21, pp. 7386–7398, 2015. <https://doi.org/10.1016/j.eswa.2015.05.027>
- [14] H. Ma, D. Zhou, C. Liu, M. R. Lyu, and I. King, “Recommender Systems with Social Regularization Categories and Subject Descriptors,” *Proc. fourth ACM Int. Conf. Web search data Min.*, pp. 287–296, 2011, <https://doi.org/10.1145/1935826.1935877>
- [15] T. Zhou, L. Lü, and Y.-C. Zhang, “Predicting missing links via local information,” *Eur. Phys. J. B*, vol. 71, no. 4, pp. 623–630, 2009. <https://doi.org/10.1140/epjb/e2009-00335-8>
- [16] Y. Koren, P. Ave, F. Park, H. D. Management, and D. Applications, “Factorization meets the neighborhood: a multifaceted collaborative filtering model,” *Proc. 14th ACM SIGKDD Int. Conf. Knowl. Discov. data Min.*, pp. 426–434, 2008, <https://doi.org/10.1145/1401890.1401944>
- [17] F. Ricci, L. Rokach, and B. Shapira, *Recommender Systems Handbook*, 2nd ed. Springer Publishing Company, Incorporated, 2015.
- [18] Z. Wu and Y. Li, “Link prediction based on multi-steps resource allocation,” in 2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2014, vol. 1, pp. 355–360. <https://doi.org/10.1109/wi-iat.2014.56>
- [19] R. Chen, Q. Hua, Y.-S. Chang, B. Wang, L. Zhang, and X. Kong, “A survey of collaborative filtering-based recommender systems: From traditional methods to hybrid methods based on social networks,” *IEEE Access*, vol. 6, pp. 64301–64320, 2018. <https://doi.org/10.1109/access.2018.2877208>
- [20] P. Moradi, S. Ahmadian, and F. Akhlaghian, “An effective trust-based recommendation method using a novel graph clustering algorithm,” *Phys. A Stat. Mech. its Appl.*, vol. 436, pp. 462–481, 2015. <https://doi.org/10.1016/j.physa.2015.05.008>
- [21] I. Cantador, P. Brusilovsky, and T. Kuflik, “Second workshop on information heterogeneity and fusion in recommender systems (HetRec2011),” in *Proceedings of the fifth ACM conference on Recommender systems*, 2011, pp. 387–388. <https://doi.org/10.1145/2043932.2044016>
- [22] J. Liu, C. Wu, and W. Liu, “Bayesian probabilistic matrix factorization with social relations and item contents for recommendation,” *Decis. Support Syst.*, vol. 55, no. 3, pp. 838–850, 2013. <https://doi.org/10.1016/j.dss.2013.04.002>
- [23] J. Tang, H. Gao, H. Liu, and A. Das Sarma, “eTrust: Understanding trust evolution in an online world,” in *Proceedings of the 18th ACM SIGKDD international conference on*

- Knowledge discovery and data mining*, 2012, pp. 253–261. <https://doi.org/10.1145/2339530.2339574>
- [24] G. Guo, J. Zhang, and N. Yorke-Smith, “A novel bayesian similarity measure for recommender systems,” in *Twenty-Third International Joint Conference on Artificial Intelligence*, 2013.
- [25] H. Ma, I. King, and M. R. Lyu, “Learning to recommend with social trust ensemble,” in *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, 2009, pp. 203–210. <https://doi.org/10.1145/1571941.1571978>
- [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*, 2008, pp. 880–887. <https://doi.org/10.1145/1390156.1390267>
- [27] Y. Koren and R. Bell, “Advances in collaborative filtering,” in *Recommender systems handbook*, Springer, 2015, pp. 77–118. https://doi.org/10.1007/978-1-4899-7637-6_3
- [28] Y. Bo, L. Yu, L. Dayou, and L. Jiming, “Social collaborative filtering by trust,” in *International Joint Conference on Artificial Intelligence AAAI Press*, 2013, pp. 2747–2753.
- [29] W. Shi, L. Wang, and J. Qin, “User Embedding for Rating Prediction in SVD++-Based Collaborative Filtering,” *Symmetry (Basel)*, vol. 12, no. 1, p. 121, 2020. <https://doi.org/10.3390/sym12010121>

6 Authors

Ali M. Ahmed Al-Sabaawi is a Ph.D. student of Electrical Electronic and Computer Engineering, Aksaray University. He received his MSc degree in computer science, at Universiti Teknologi Malaysia in 2013. His research interested lie in the areas of data mining techniques and machine learning applications. Email: alisabaawi5@gmail.com

Hacer Karacan is an Assoc. Prof. of Computer Engineering at the Gazi University, Turkey. She received a Ph.D. degree in Cognitive Science from the Middle East Technical University in 2007. Her research interests lie in the areas of artificial intelligence, interactive system design, machine learning applications and data clustering.

Yusuf E. Yenice is an Assist Prof. of Electrical Electronic and Computer Engineering at Aksaray University. He received his MSc and PhD degrees in electronic engineering from University of Surrey, United Kingdom, in 1994 and 1999, respectively. His research interest is free-space optical communications and data mining techniques.

Article submitted 2020-08-13. Resubmitted 2020-10-04. Final acceptance 2020-10-07. Final version published as submitted by the authors.