Matrix Factorization Based Recommendation Algorithm for Sharing Patent Resource

Xueqing ZHANG\(^{(a)}\), Xiaoxia LIU\(^{(b)}\), Jun GUO\(^{(c)}\), Nonmembers, Wenlei BAI\(^{(d)}\), Student Member, and DAGUANG GAN\(^{(e)}\), Nonmember

SUMMARY As scientific and technological resources are experiencing information overload, it is quite expensive to find resources that users are interested in exactly. The personalized recommendation system is a good candidate to solve this problem, but data sparseness and the cold starting problem still prevent the application of the recommendation system. Sparse data affects the quality of the similarity measurement and consequently the quality of the recommender system. In this paper, we propose a matrix factorization recommendation algorithm based on similarity calculation (SCMF), which introduces potential similarity relationships to solve the problem of data sparseness. A penalty factor is adopted in the latent item similarity matrix calculation to capture more real relationships furthermore. We compared our approach with other 6 recommendation algorithms and conducted experiments on 5 public data sets. According to the experimental results, the recommendation precision can improve by 2% to 9% versus the traditional best algorithm. As for sparse data sets, the prediction accuracy can also improve by 0.17% to 18%. Besides, our approach was applied to patent resource exploitation provided by the wanfang patents retrieval system. Experimental results show that our method performs better than commonly used algorithms, especially under the cold starting condition.

key words: scientific and technological resources recommendation, recommendation systems, matrix factorization, potential similarity matrix, data sparseness

1. Introduction

With the explosive growth of a wide variety of scientific and technological resources, it is difficult for users to find a certain type of resource on data service platforms. Since resources can link by references or keywords, it is the most feasible way to implement more accurate resource recommendations based on users’ retrieval information. The personalized recommendation mechanism plays an important role in scientific and technological resources exploitation.

In the past decade, exceptional progress has been achieved in personalized recommendation systems, which extracts potential characteristics of users and items based on users’ ratings information, and then recommends different items that may be of interest to users. Among these personalized recommendation algorithms, matrix factorization [1]–[7] attracts more attention in the academic and industrial communities due to its superior theoretical foundation and strong stability. With the development of trust networks of users, the social recommendation mechanism [6], [8]–[13] applies to recommendation systems. Experimental results show that this mechanism can not only improve accuracy but also effectively alleviate the data sparseness. Simultaneously, we suggest considering the associated relationships of items while calculating similarities of each item.

In this paper, we proposed a matrix factorization recommendation algorithm based on similarity calculation (SCMF). To prevent over-fitting, we added regularization terms to the objective function. The algorithm also added user bias, item bias, and global bias to improve prediction accuracy. Moreover, both the social relationships of users and associated relationships of items are introduced to the model to solve the data sparseness [14] and improve the prediction accuracy. Since each item has a different chance of being rated, a penalty factor is adopted to obtain real similarities of items. Finally, we applied our algorithm to a scientific and technological resource provided by the wanfang Data Co.. Tremendous experiments verified the efficiency of our method.

2. Related Work

2.1 Collaborative Filtering Algorithm

Collaborative filtering (CF) is the most widely used recommendation algorithm for items recommendation [8], [15]–[19]. It has the advantages of strong generality, not requiring much expertise, and simple implementation. As a practical recommendation algorithm, it prompts personalized recommendations tremendously. The objective of collaborative filtering is to calculate users’ similarities by the ratings of the same items and then recommend the same interest user preferences items. However, sparse data sets have fewer ratings for the same items, and it is impossible to obtain the exact interest preferences of users. So the recommendation performance is usually not satisfying in this case. Especially for newly registered users, the algorithm can not make personalized recommendations for new users on the existing ratings, which is called the cold starting problem [14].
2.2 Matrix Factorization Algorithm

In content-based recommendation methods, the traditional matrix factorization algorithm is widely adopted, which uses the matrix factorization function to decompose the user-item rating matrix into the potential feature matrices of users and items (to facilitate the following, we call the algorithm as SVD). By producing the two feature matrices, the algorithm can predict the unknown ratings of the user-item matrix. The traditional matrix factorization method has good expansibility. It can set the step of the feature dimension according to the demand. The algorithm can also generate potential feature vectors for sparse data sets or newly registered users and makes it possible to provide personalized recommendations.

Meanwhile, ratings in each data set are also affected by the characteristics of users or items themselves. For example, for tolerant users, their ratings are generally friendly and much higher. While for rigorous users, their ratings are usually much lower. And for items, taking article resources as an example, articles published in high-level conferences or journals may get higher recognition. While in ordinary conferences or journals, their recognition may slightly decline. These factors are independent of users and items, and there is nothing to do with users’ preference for items. Therefore, based on the traditional matrix factorization algorithm, the algorithm adds user bias and item bias to achieve personalized recommendations on each data set. There are distinctions in different rating data sets, so it is necessary to add a global bias to the algorithm to reflect the implicit feedback information. Therefore, matrix factorization with bias model [2], [12] proposes (to facilitate the following, we call this algorithm as SVD++). Experimental results show that this recommendation algorithm has the best performance among the existing popular algorithms.

2.3 Co-Regulation Mechanism Algorithm

The co-regularization mechanism matrix factorization algorithm [3], [20], [21] is a much newer personalized recommendation method. Based on the traditional matrix factorization, the model adds regularization terms to avoid overfitting. The social matrix of users and items’ associated matrix are applied to capture real relationships and alleviate the insufficient data information. Finally, the model adds user bias and item bias to improve the prediction accuracy (to facilitate the following, we call the algorithm as Co-R). The algorithm has experimentally proved that its prediction accuracy is significantly better than the traditional matrix factorization algorithm SVD.

3. Construction of SCMF Algorithm

3.1 Social Relationships of Users

The users’ similarities matrix describes the social relationships of users. Pearson correlation coefficient and cosine similarity are universal methods for measuring similarities of users and items. In this paper, we use the cosine similarity calculation method. However, this method does not take the differences in rating scale between different users into account. The adjusted cosine similarity [16] can offset this drawback by subtracting the corresponding user average from each co-rated pair. Inspired by this method, we consider the rating habit of each user. Some users are more friendly and generally give higher ratings, while some users are more picky and likely to give lower ratings. This rating habit does exist, and it is also easy to dig. The modified cosine similarity calculation method can offset the rating habits of different users. Compared to the adjusted cosine similarity calculation method, the experimental results can prove the modified cosine similarity calculation method’s effectiveness. So in this paper, the modified cosine similarity is adopted to obtain similarities of users. The calculation equation can describe as following:

\[
S_{uv} = \frac{\sum_{x \in (B_u \cap B_v)} (R_{ux} - \bar{R}_u)(R_{vx} - \bar{R}_v)}{\sqrt{\sum_{x \in (B_u \cap B_v)} (R_{ux} - \bar{R}_u)^2} \times \sqrt{\sum_{x \in (B_u \cap B_v)} (R_{vx} - \bar{R}_v)^2}}
\]

Among them, \(S_{uv}\) denotes the similarity between user \(u\) and user \(v\); \(B_u\) and \(B_v\) are ratings set of users \(u\) and \(v\) in the training set; \(\bar{R}_u\) and \(\bar{R}_v\) are average ratings of users \(u\) and \(v\) in the training set; the matrix \(S\) is users’ social relationships.

3.2 Associated Relationships of Items with Penalty

Similar to the social relationships, the associated relationships of items can be expressed by items’ modified cosine similarity matrix. However, in actual ratings, some items have a wider audience, which will bring more opportunities for users to evaluate these items. There is taking the article resource for an example. In the recommendation system, collaborative filtering methods are the theoretical foundation, so their articles are generally have high citations, and we call them positive items. The deep learning method based on graph network is the progress of this field, so citation probability of these articles reduce. These articles are called negative items. Considering their chance of being cited, we believe that there are substantial similarities between negative items.

In personalized recommendation systems, users always expect to recommend items of real interest rather than popularity. Therefore, we added a penalty function to the modified cosine similarity in obtaining the items’ associated relationships. This function weakens the contribution of positive items and improves the weight of negative items. In this case, it helps to capture the authenticity of associated relationships. Therefore, combining the modified cosine sim-
ilarity and a punishment factor for positive items, the improved items similarity calculation equation can describe as following:

\[
W_{ij} = \frac{\sum_{y \in (D_i \cap D_j)} (R_{yi} - \bar{R}_i)(R_{yj} - \bar{R}_j)}{\sqrt{\sum_{y \in (D_i \cap D_j)} (R_{yi} - \bar{R}_i)^2} \sqrt{\sum_{y \in (D_i \cap D_j)} (R_{yj} - \bar{R}_j)^2}} \times \frac{1}{\log(1+U(i))}
\]  

(2)

Among them, \(W_{ij}\) indicates the similarity between item \(i\) and item \(j\); \(D_i\) and \(D_j\) are ratings set of items \(i\) and \(j\) in the training set; \(\bar{R}_i\) and \(\bar{R}_j\) are average ratings of items \(i\) and \(j\) in the training set; \(U(i)\) indicates the total number of users who evaluate item \(i\), and the addition part in Eq. (2) is to punish the proportion of positive items in rating prediction; the final result \(W\) is the matrix of the items’ associated relationships. The associated matrix can not only obtain sincere and stable relationships but also alleviate data sparseness.

3.3 SCMF Algorithm

Based on the traditional matrix factorization model, the SCMF algorithm implements by first converting a given user-item rating data set into a \(m \times n\) high-dimensional space matrix with \(m\) users and \(n\) items, which is called \(R\). Then the method uses the matrix factorization function to decompose \(R\) into low-dimensional users’ potential feature matrix \(P\) and items’ potential feature matrix \(Q\). The unknown ratings in the user-item matrix can obtain by \(R = P_uQ_d^T\), where \(P_u\) and \(Q_d\) are feature vectors of dimension \(k\) (\(k\) is much smaller than the number of \(m\) and \(n\)). By using the known ratings and Least-square method, the loss function of the model defines as the following:

\[
L = \frac{1}{2} \sum_{u=1}^{m} \sum_{i=1}^{n} (R_{ui} - \sum_{d=1}^{k} P_{ud}Q_{id}^T)^2
\]  

(3)

According to social relevance theory, users prefer to establish connections with items that match their preferences rather than automatically recommended by product manufacturers or systems. Therefore, social and associated relationship matrices are necessary to capture users’ and items’ potential preference information. Since they have different effects on the objective function, it is required to select two parameters to adjust their weights. Regularization [11], [16] coefficients are adopted to avoid excessive reliance on training data. As the \(F\) paradigm is a generally used regularization term, the loss function can describe as:

\[
L = \frac{1}{2} \sum_{u=1}^{m} \sum_{i=1}^{n} (R_{ui} - \sum_{d=1}^{k} P_{ud}Q_{id}^T)^2 + \frac{\lambda}{2}(\|P\|_F^2 + \|Q\|_F^2) + \frac{\beta}{2} S_{av}(\|P_u - P_v\|_F^2) + \frac{\alpha}{2} W_{ij}(\|Q_i - Q_j\|_F^2)
\]  

(4)

Where \(\lambda, \beta, \alpha\) are regularization hyper-parameters; \(P\) is the users’ features matrix; \(Q\) is the items’ features matrix; \(P_u\) and \(Q_d\) are users’ feature vectors; \(Q_i\) and \(Q_j\) are items’ feature vectors.

Taking the rating characteristics of users and items into account, our model introduces global bias \(\mu\), user bias term \(b_{pu}\), and item bias term \(b_{qj}\) to eliminate the influence of rating habits. In this paper, we consider the impact of the additions of similarities on computational performance, so all bias terms and potential feature vectors in the model use the same hyper-parameters to adjust their weights. The \(2\) paradigm has the advantage of easy computing and often calculates regularized terms. In summary, the final loss function describes as follows:

\[
L = \frac{1}{2} \sum_{u=1}^{m} \sum_{i=1}^{n} (R_{ui} - b_{pu} - b_{qj} - \mu - \sum_{d=1}^{k} P_{ud}Q_{id}^T)^2 + \frac{\lambda}{2}(\|P\|_F^2 + \|Q\|_F^2 + \|b_{pu} - b_{qj}\|_F^2) + \frac{\beta}{2} S_{av}(\|P_u - P_v\|_F^2) + \frac{\alpha}{2} W_{ij}(\|Q_i - Q_j\|_F^2)
\]  

(5)

Gradient descent is a generally introduced optimization method [22] to find the minimum of the loss function. The gradient descent method has the advantages of low initial point requirements and fewer storage variables. Since this method calculates all sample data, it has the disadvantages of slow convergence and low efficiency. The random gradient descent algorithm is an alternative to gradient descent. It only randomly selects one sample of data to calculate the gradient instead of all data, which improves the calculation efficiency. Therefore, this paper applies stochastic gradient descent to solve the loss function. In the training process, the algorithm performs gradient descent on each parameter to minimize the gradient of Eq. (5), the updated values introduce to calculate the new objective function. All of these steps iterate until the loss function \(L\) converges. There is a description of the SCMF algorithm.

4. Experimental Results and Analysis

4.1 Data Sets

Taking the computational cost of the matrix and the sparseness of ratings into account, we applied 5 public item and movie data sets in the recommendation system for experiments: MovieLens100K, Movielens1M, Filimtrust, Epinions, and Ciao. In MovieLens100K, 70 users rate 3952 movies, and in Movielens1M, 774 users rate 3952 movies. All these users can rate movies in the range of \([1, 5]\) with a step size of 1. In Filmtrust, users can rate movies in the range of \([0.5, 4]\) according to their preferences, with a step size of 0.5. We selected 102 users to evaluate 9014 items in Epinions. The evaluation range is \([1, 5]\), and the step size is 1. In the Ciao movies data set, the users’ evaluation range is
Algorithm 1 Description of SCMF algorithm.

Input: Training set \( R^T \); social relationships matrix of users \( S \); associated relationships matrix of items \( W \); latent feature dimension \( d \); hyper-parameters \( \lambda, \beta, \alpha, \gamma \); learning rate \( \gamma \); iteration numbers \( steps \).

Output: Users latent features matrix \( P \); items latent features matrix \( Q \); user bias \( b_u \); item bias \( b_i \); P and Q.

1. Randomly initial \( b_p, b_q, P \) and \( Q \).
2. FOR \( i \) from 1 : \( |R^T| \) DO
3. \( eui = R_{ui} - b_u - b_i - \mu - \sum_{d=1}^{\text{d}} P_{ud} Q_{id}^T \)
4. \( b_{pu} = b_{pu} + \gamma (eui - \lambda \times b_{pu}) \)
5. \( b_{qi} = b_{qi} + \gamma (eui - \lambda \times b_{qi}) \)
6. \( temp_p = p_u \)
7. FOR \( e \) from 1 : \( m \) DO
8. \( tmp = temp + S_{we} \times (P_u - P_e) \)
9. END
10. \( P_u = P_u + \gamma (eui + \beta \times temp - \alpha \times b_{pu}) \)
11. FOR \( i \) from 1 : \( n \) DO
12. \( temp = temp + W_{ei} \times (Q_i - Q_e) \)
13. END
14. \( Q_i = Q_i + \gamma (eui + \beta \times temp - \alpha \times b_{qi}) \)
15. \( \gamma = 0.93 \times \gamma \)
16. END
17. UNTIL convergence
18. return \( b_p, b_q, P, Q \);

Table 1 Feature statistics of 5 public data sets.

| MovieLens | Epinions | Filmenjoy | Cho2011 |
|-----------|----------|-----------|---------|
| Users     | 70,774   | 102       | 1,508   | 7,858   |
| Items     | 3,952    | 9,014     | 2,071   | 5,166   |
| Ratings   | 10k/120k | 10k       | 35,497  | 20k     |
| Density   | 3.61%    | 3.92%     | 1.09%   | 1.14%   | 0.049%  |
| Quantum   | user<item>| user<item>| user<item>| user<item>|

4.2 Criterion

For the recommendation performance of models, there are lots of deterministic evaluation indexes to measure them. We adopt two evaluation standards in recommendation algorithms: root mean squared error (RMSE) and mean absolute error (MAE) [21]. They widely apply to accurate rating prediction, and the calculation methods of the two indexes indicate in Eq. (6) and Eq. (7). The accuracy of recommendation results is measured by calculating the difference between real ratings and predict ratings. The smaller the values are, the higher the recommendation accuracy of the algorithm is.

\[
RMSE = \sqrt{\frac{1}{|R^T|} \sum_{(u,i) \in R^T} (R_{ui} - \hat{R}_{ui})^2} \tag{6}
\]

\[
MAE = \frac{1}{|R^T|} \sum_{(u,i) \in R^T} |R_{ui} - \hat{R}_{ui}| \tag{7}
\]

Where \( R^T \) is the testing set and \( |R^T| \) is the number of ratings in the testing set.

4.3 Experiment Setting

The following is a detailed description of 6 recommendation algorithms:

UCF: User-based collaborative filtering, this method is to find similar users and recommend their favorite items.

ICF: Item-based collaborative filtering, which is to find similar items for recommendation. The number of UCF and Matrix factorization algorithm is all set to 10.

SVD: The traditional matrix factorization algorithm, and it only considers users’ rating information for items.

SVD++: A Matrix factorization algorithm that considers user bias, item bias, and global bias. There is a high accuracy recommended algorithm in matrix factorization methods.

Co-R: The Co-regularization mechanism matrix factorization considers the similarity of users and items, user bias, item bias, and penalties for active users, but ignores the impact of global bias on the performance of recommendations. It is the newer algorithm of matrix factorization so far.

SCMF: This is our algorithm, and it considers the similarity of users, similarity of items, user bias, item bias, and also considers the impact of positive items on the calculation of similarities of items.

SCMF forpunished: The algorithm is generally consistent with SCMF. However, it does not consider the punishment of positive items when calculating the similarities. This algorithm aims to compare with the SCMF algorithm and to prove the effectiveness of the penalty.

Based on experience, we select the relatively optimal parameter interval by halving the numerical range. Finally, we set hyper-parameters to 1, 0.1, 0.01, and 0.001, and latent features vector dimension \( k \) is 10. The specific parameters of each algorithm for each data set indicate in Table 2. And the following experimental results and analysis are obtained with 5 data sets under relatively optimal parameters.

In all experiments, each data set can divide into the training set and the testing set. The training set aims to learn the parameters of models, and then models use the testing set to evaluate the accuracy of algorithms. Each data set will randomly choose 80% as the training set and 20% as the testing set.

4.4 Experimental Results and Analysis

In this subsection, we aim to compare our proposed algorithm with representative recommendation algorithms and evaluate its effectiveness furthermore.

Firstly, we compare these recommendation algorithms on 5 data sets. Table 3 and Table 4 show the experimental results of RMSE and MAE. From the experimental results, the prediction accuracy only based on UCF or ICF is not satisfying. SVD and SVD++ algorithms have a great improvement compared with UCF and ICF, which illustrates
the effectiveness of matrix factorization methods. Co-R has an excellent recommendation performance compared with UCF and ICF. The enormous improvement indicates that the addition of users’ and items’ relationships in the matrix factorization model can improve accuracy. Compared with Co-R, the SCMF has a significant improvement in recommendation performance, and it can explain that global bias information helps to improve the prediction of personalized recommendations.

Secondly, we evaluate the performance of the recommendation algorithms on sparse data sets. We select users whose ratings less than 5 items and 10 items, and the users’ rating information applies as a new training set. The corresponding algorithms are learning on this training set. Their RMSE and MAE get calculated on the testing set, and detailed results show in Fig. 1 and Fig. 2.

In the two data sets of MovieLens, there are no users with ratings less than 5 items on the randomly selected training set. There are also no users with ratings less than 10 items on the randomly selected training set of Movielens1M data set. It is equivalent to the absence of users’ rating information on the new training set, and recommendation algorithms cannot learn parameters on the training set, so performance evaluation indicators on 6 recommendation algorithms are 0.

From the experimental results of other data sets, the SCMF algorithm is significantly better than mainstream algorithms. Since the rating information is sparse, social relationships of users are often sparse. At the same time, the associated relationships of items can improve the recommendation accuracy of this algorithm. The Co-R algorithm applies the items’ associated relationships but ignores the global information of users’ ratings. As a result, the prediction accuracy of the recommendation is significantly lower than the SCMF.

We also do experiments on data sets with users’ ratings less than 10, and compared with data sets with users’ ratings less than 5, both RMSE and MAE show downward trends. As the number of ratings increasing, data density becomes relatively higher, considering both social relationships of users and associated relationships of items can alleviate this sparseness. In this way, it improves the prediction accuracy of the algorithm.

### 4.5 Application Example

Finally, we applied our algorithm to the wanfang patent data set as an application of resource recommendation. There are 11483 patents in the wanfang patents, and the data set covers from 1985 to 2018. Each patent record consists of patent classification, patent number, applicant, inventor, related search terms, and so on. Up to now, there is no clear rating information in the resource data information. To be

---

**Table 2** The specific parameters of each algorithm for each data set.

| Data      | MovieLens100K | MovieLens1M | Epinions | Filmtrust | Ciao  |
|-----------|---------------|-------------|----------|-----------|-------|
| UCF       | neighbors k=10| neighbors k=10| neighbors k=10| neighbors k=10| neighbors k=10 |
| ICF       | neighbors k=10| neighbors k=10| neighbors k=10| neighbors k=10| neighbors k=10 |
| SVD       | gamma=0.01, t=0.1 | gamma=0.01, t=0.01 | gamma=0.01, t=0.01 | gamma=0.01, t=0.1 | gamma=0.01, t=0.1 |
| SVD++     | gamma=0.01, t=1  | gamma=0.001, t=0.01 | gamma=0.001, t=0.01 | gamma=0.001, t=0.1 | gamma=0.001, t=0.1 |
| Co-R      | gamma=0.01, t=β=α=0.1 | gamma=β=0.01, t=0.01 | gamma=α=0.01, t=0.01 | gamma=α=0.01, t=0.01 | gamma=α=0.01, t=0.01 |
| SCMF      | gamma=0.01, t=β=α=0.1 | gamma=β=0.01, t=0.01 | gamma=α=0.01, t=0.01 | gamma=α=0.01, t=0.01 | gamma=α=0.01, t=0.01 |

---

**Table 3** RMSE comparison on different algorithms.

| Data       | MovieLens100K | MovieLens1M | Epinions | Filmtrust | Ciao  |
|------------|---------------|-------------|----------|-----------|-------|
| UCF        | 3.1132        | 2.8952      | 3.5779   | 2.2338    | 3.6193 |
| ICF        | 2.9225        | 2.8272      | 4.1195   | 2.2273    | 3.4620 |
| SVD        | 0.7645        | 0.5280      | 1.5751   | 0.8695    | 1.2111 |
| SVD++      | 0.5219        | 0.4947      | 0.5164   | 0.4517    | 0.5125 |
| Co-R       | 0.5973        | 0.5199      | 0.8577   | 0.5419    | 1.2957 |
| SCMF       | 0.4828        | 0.4753      | 0.5019   | 0.4095    | 0.4764 |
| Improvement| 0.0391        | 0.0194      | 0.0145   | 0.0422    | 0.0361 |

---

**Table 4** MAE comparison on different algorithms.

| Data       | MovieLens100K | MovieLens1M | Epinions | Filmtrust | Ciao  |
|------------|---------------|-------------|----------|-----------|-------|
| UCF        | 2.9169        | 2.6740      | 3.4182   | 2.0685    | 3.4237 |
| ICF        | 2.7287        | 2.5968      | 3.9624   | 2.0450    | 3.2975 |
| SVD        | 0.3015        | 0.2084      | 0.7291   | 0.3859    | 0.4870 |
| SVD++      | 0.2107        | 0.1995      | 0.1931   | 0.1764    | 0.1997 |
| Co-R       | 0.2385        | 0.2097      | 0.3675   | 0.2089    | 0.5667 |
| SCMF       | 0.1915        | 0.1891      | 0.1891   | 0.1607    | 0.1849 |
| Improvement| 0.0192        | 0.0102      | 0.0004   | 0.0157    | 0.0148 |
Table 5 RMSE comparison on wanfang patents data.

| Data     | wanfang patent | less 1 patents | less 2 patents | less 1 applicant | less 2 applicants |
|----------|----------------|----------------|----------------|------------------|------------------|
| UCF      | 3.1650         | 3.3850         | 3.7746         | —                | —                |
| ICF      | 2.9151         | —              | —              | 2.7069           | 2.8243           |
| SVD      | 0.8963         | 3.0652         | 2.8319         | 2.7040           | 2.8043           |
| SVD++    | 0.5146         | 1.4089         | 1.2481         | 1.0773           | —                |
| Co-R     | 0.6274         | 1.6826         | 1.3894         | 1.3002           | 1.1673           |
| SCMF     | 0.5044         | 1.5220         | 1.1803         | 1.0465           | 0.9943           |
| Improvement | 0.0102     | 0.0769         | 0.0678         | 0.0826           | 0.0830           |

Table 6 MAE comparison on wanfang patents data.

| Data     | wanfang patent | less 1 patents | less 2 patents | less 1 applicant | less 2 applicants |
|----------|----------------|----------------|----------------|------------------|------------------|
| UCF      | 2.9642         | 3.7019         | 3.6851         | —                | —                |
| ICF      | 2.7003         | —              | —              | 2.5456           | 2.6067           |
| SVD      | 0.3716         | 2.7222         | 2.6651         | 2.4913           | 2.5867           |
| SVD++    | 0.2081         | 1.1057         | 1.0577         | 0.8820           | 0.8470           |
| Co-R     | 0.2559         | 1.3112         | 1.1927         | 1.0425           | 0.9287           |
| SCMF     | 0.2026         | 0.9847         | 0.8327         | 0.8442           | 0.8046           |
| Improvement | 0.0055     | 0.1210         | 0.2250         | 0.0378           | 0.0424           |

applied to the recommendation algorithm, it is necessary to manually rank patents within the range of [1,5] based on the publication date. We also need to number the patent numbers and applicants mentioned in the record, and finally, construct a patent number-applicant rating matrix.

The patents’ relationships matrix can use the applicant, inventor, application date, patent classification, and the number of identical related search terms to calculate similarities. For the applicants’ relationships matrix, we use the application date, patent classification, and the number of the same related search terms to calculate similarities. By adjusting the relatively best hyper-parameters, Table 5 and Table 6 show the experimental results.

By observing the experimental results, the SCMF algorithm also efficiently improves prediction accuracy by 2% to 2.7% on the wanfang patents data compared with the SVD++ algorithm. Analogous to sparse user data sets, we also select patents’ sparseness data sets with applied less than 1 and 2, and applicants’ sparseness data sets with applicant number is less than 1 and 2. From the results, we can see that our algorithm also has good recommendation results. Its performance can improve by 2% to 7% compared with the best mainstream algorithm SVD++.

Besides, comparing the data set with less than 2 patent applications with the data set with less than 1 patent application, the results of RMSE and MAE reduce. It can interpret when the interaction information increases, our algorithm can capture more potential real relationships and then make more accurate and personalized recommendations. It is also consistent with the conclusion of item recommendations.

4.6 Influence of Hyper-Parameters on SCMF

The Filmtrust data set has the best performance in the 5 data sets, and its density of users’ ratings for items is similar to many actual rating data sets. So we selected the Filmtrust to analyze the influence of learning rate, bias information, social relationships of users, and associated relationships of items on the performance of the SCMF. Figure 3 shows the results.

Hyper-parameter \( \lambda \) controls the proportion of all bias information in the model. As the value increases, its tendency first decreases, it increases when reaching a certain threshold and then gradually decreases. It shows that the proportion of bias information affects the quality of recommendation. Especially, when the \( \lambda \) value is 0.01, its RMSE value is significantly smaller than the \( \lambda \) value is 0, which illustrates the effectiveness of adding bias information.

Hyper-parameters \( \beta \) and \( \alpha \) respectively control the proportion of social relationships of users and associated relationships of items. The figure shows the effect of changing hyper-parameters on RMSE. As value increases, the \( \beta \) slightly decreases, rises after reaching a certain threshold, and then stabilizes. Comparing with the case where there are no social relationships of users, the recommendation accuracy is slightly improved when the \( \beta \) is 0.1. This result shows that the proportion of social relationships of users in the SCMF algorithm directly affects the recommendation quality. As the \( \alpha \) value increases, the SCMF algorithm recommendation accuracy increases at first, then decreases, and finally increases. When the value of \( \alpha \) is 0.01, the rating prediction accuracy is higher than the value of 0, which illustrates the effectiveness of adding associated relationships of items furthermore.

4.7 Influence of Features Dimension \( k \) on SCMF

To observe the effect of feature dimension on prediction performance during matrix factorization, we set the number of features to 5, 10, 15, and 20. These dimensions reference
a related paper [20] of previous work. There are some experiments to make comparisons. The results show that this setting is reasonable, and its recommendation accuracy is satisfying. We use our proposed SCMF algorithm to make predictions on the Filmtrust data set, and Table 7 shows the prediction results.

Theoretically, the higher the dimensionality of the features is, the better the prediction performance is. However, when a computer calculates the dot product, the floating-point number only retains a limited number of digits, and the result needs to sum. These processes will produce larger errors with the increase of attribute characteristics. The above experiments show that when the feature dimension is 10, there is a relatively superior prediction and running rate.

5. Conclusion

In this paper, we propose the SCMF algorithm to alleviate the cold starting problem. In contrast with mainstream methods, our algorithm considers both users’ social matrix and items’ associated matrix to capture more potential information. To further improve the recommendation accuracy, the SCMF algorithm also adds a penalty factor of positive items. Besides, we successfully apply our algorithm to recommend a scientific and technological resource. Analogous to the social relationships and associated relationships, the SCMF utilizes the patents’ and applicants’ relationships to capture more potential similarity information. As for evaluating excessive experimental results indicate that the SCMF algorithm is also suitable for the recommendation of scientific and technological resources.

However, our SCMF algorithm also has a shortcoming. The efficiency gets improved by using the matrix factorization function but reduces when calculating similarities of users and items. In the next work, two aspects can be advanced. Considering the search efficiency of similar neighbors, we will introduce the local sensitive hash method. The deep learning of the graph network can improve the expression ability of feature vectors, which will make it possible to improve the accuracy of prediction.

Acknowledgments

This work is financially supported in part by the National Key Research and Development Program of China under Grant No.2017YFB1400301 and the National Natural Science Foundation of China Project under Grant No.62072372, 61973250, 61902318, and the Key Research and Development Program of Shaanxi Province under Grant 2019GY-012.

Table 7 Results comparison with different features dimension.

| Data   | Criterion | $k = 5$ | $k = 10$ | $k = 15$ | $k = 20$ |
|--------|-----------|---------|----------|----------|----------|
| Filmtrust | RMSE      | 0.8441  | 0.8175   | 0.8193   | 0.8360   |
|         | MAE       | 0.6764  | 0.6480   | 0.6553   | 0.6613   |

References

[1] Y. Koren, R. Bell, and C. Volinsky, “Matrix factorization techniques for recommender systems,” Computer, vol.42, no.8, pp.30–37, Aug. 2009.
[2] R.X. Song and G.Y. Li, “Application of improved matrix factorization model in recommendation system,” Computer Applications, vol.39, no.1, pp.721–731, 2019.
[3] X. Yu and H. Zhang, “Low-rank matrix factorization recommendation model based on double regularization mechanism,” Application Research of Computers, April 2020.
[4] G. Zhang and X. Zhou, “Similarity based matrix factorization for recommender systems,” 10th Int. Symp. Computational Intelligence and Design, pp.7–11, 2017.
[5] P. Pirasteh, D. Hwang, and J.J. Jung, “Exploiting matrix factorization to asymmetric user similarities in recommendation systems,” Knowledge Based Systems, vol.83, pp.51–57, March 2015.
[6] H. Ma, H. Yang, M.R. Lyu, and I. King, “Sorec: social recommendation using probabilistic matrix factorization,” 17th ACM Conf. Information and Knowledge Management, pp.931–940, Oct. 2008.
[7] A. Mnih and R. Salakhutdinov, “Probabilistic matrix factorization,” Int. Conf. Neural Information Processing Systems, pp.1257–1264, 2007.
[8] F.B. Kharrat, A. Elkhlifi, and R. Faiz, “Empirical study of social collaborative filtering algorithm,” Proc. Asian Conf. Intelligent Information & Database Systems, pp.85–95, March 2016.
[9] Y. Yang, Y. Guo, Y. Liu, and H. Steck, “A survey of collaborative filtering based social recommender systems,” Computer Communications, vol.41, pp.1–10, March 2014.
[10] M. Jamali and M. Ester, “Using a trust network to improve top-n recommendation,” 3th ACM Conf. Recommender Systems, pp.181–188, Oct. 2009.
[11] B. Pal and M. Jenamani, “Trust inference using implicit influence and projected user network for item recommendation,” J. Intelligent Information Systems, vol.52, no.2, pp.425–450, June 2019.
[12] T. Chen, Q. Zhu, and M.X. Zhou, “Trust-based recommendation algorithm in social network,” Software, vol.28, no.3, pp.93–95, 2017.
[13] M. Jamali and M. Ester, “A matrix factorization technique with trust propagation for recommendation in social networks,” Proc. 4th ACM Conf. Recommender System, pp.135–142, Jan. 2010.
[14] S. Sedhain, A.K. Menon, S. Sanner, L. Xie, and D. Brazianas, “Low-rank linear cold-start recommendation from social data,” Proc. 31th ACM Conf. Information and Artificial Intelligence, pp.1502–1508, Feb. 2017.
[15] P. Massa and P. Avesani, Trust-Aware Collaborative Filtering for Recommender Systems, Oct. 2004.
[16] B.M. Sarwar, G. Karypis, J.A. Konstan, and J. Riedl, “Item-based collaborative filtering recommendation algorithms,” Proc. ACM Conf. The Special Interest Group on Data Communication, pp.285–295, Aug. 2001.
[17] J. Tang, S. Wang, X. Hu, D. Yin, Y. Bi, Y. Chang, and H. Liu, “Recommendation with social dimensions,” Proc. 31th AAAI Conf. Information and Artificial Intelligence, pp.1502–1508, Feb. 2017.
[18] B. Wu, Z.Z. Lou, and D.Y. Ye, “Co-regulation matrix factorization recommendation algorithm,” Software, vol.29, no.9, pp.2681–2696, 2018.
[19] H. Ma, D. Zhou, C. Liu, M.R. Lyu, and I. King, “Recommender systems with social regularization,” Proc. 4th ACM Conf. Web Search and Data Mining, pp.287–296, Feb. 2011.
[22] L. Xiang, Recommendation System, Posts & Telecom Press, Beijing, 2012.

**Xueqing Zhang** is a postgraduate in School of Information Science and Technology, Northwest University, Xi’an, China. She received a B.E. with a major in computer science and technology at Northwest University, in 2019. Her research interests are recommendation systems and machine learning.

**Xiaoxia Liu** is a professor in the School of Information Science and Technology, Northwest University, Xi’an, P.R.China. Her research interests are intelligent information system, image processing and computer science/engineering education.

**Jun Guo** is a professor in the Department of Computer Science at Northwest University, China. He got his Ph.D. degree in computer science at Northwest University in 2007 and worked as a post-doctoral fellow in Tsinghua University during 2008–2010. His research interests are embedded computing, natural language processing and AI algorithm.

**Wenlei Bai** received the B.E. degree in mathematics and Computer Science from Shanxi Normal University, Linfen, China, in 2019. He is currently a Master’s degree candidate in Software engineering at Northwest University, China. His research interests include machine learning and recommendation systems.

**Daguang Gan** received the M.S. degree in computer science from the institute of science and technology, P.R. China, in 2008. He is an assistant to the general manager of Beijing Wan Fang software co., LTD.. His research interests include knowledge organization and information recommendation.