International Journal of Information Technologies and Systems Approach
Volume 16 • Issue 3

Research on Singular Value Decomposition Recommendation Algorithm Based on Data Filling

Yarong Liu, Guangxi Key Laboratory of Embedded Technology and Intelligent System, Guilin University of Technology, China

Feiyang Huang, School of Information Science and Engineering, Guilin University of Technology, China

Xiaolan Xie, Guangxi Key Laboratory of Embedded Technology and Intelligent System, Guilin University of Technology, China*

Haibin Huang, School of Information Science and Engineering, Guilin University of Technology, China

ABSTRACT

In the era of big data, the problem of information overload has become increasingly prominent. Recommendation systems are widely studied due to the problem. Due to the sparseness of data, the recommendation effect is not always ideal. To alleviate the problem of data sparsity, a singular value decomposition recommendation algorithm based on data filling is proposed. First, an improved Tanimoto similarity coefficient calculation method is proposed to calculate the similarity, and effective interpolation data is generated for the singular value decomposition model according to the proposed prediction formula. The experimental results show that when using the same dataset MovieLens100K, compared with several commonly used recommendation algorithms, the improved algorithm improves the prediction accuracy of the model, In the best case, RMSE is 10.1% lower than KNNBasic, 7.8% lower than Slope One algorithm, 6.9% lower than SVD algorithm, and 4.8% lower than SVD++ algorithm, verifying that this method can improve the recommendation quality.

KEYWORDS

Big Data, Data Filling, Data Sparse, Recommendation Algorithm, Similarity Calculation, Singular Value Decomposition

1 INTRODUCTION

In recent years, a new round of information technologies represented by big data and artificial intelligence has had a huge impact on our lives. Compared with traditional shopping and information acquisition methods, people prefer the Internet. However, the data on the network is extremely large, so it is difficult to find the information that users are really interested in, and the recommendation system

DOI: 10.4018/IJITSA.320222 *Corresponding Author

came into being. The recommendation system recommends to users according to their preferences. Although there are various recommendation methods, the accuracy of recommendation is not ideal. Therefore, how to improve the recommendation quality is still the focus of most researchers.

The existing recommendation methods mainly include collaborative filtering(CF) (Jia, J., Liu, P., & Chen, W., 2021), content-based filtering (Mohammadi, M., Naree, S. A., & Lati, M., 2020), hybrid recommendation (Chen, Y., & Wang, Y., 2020) and others. The CF algorithm is widely used in recommendation system (Luo, X., Xia Y., & Zhu, Q., 2012), but there are still deficiencies in these methods at present. For examples: CF algorithm has both the sparsity problem and the cold start problem (Wei, J. et al.,2017; Guo, X. et al., 2019); Content-based approaches require additional information, which is not always provided; Hybrid methods also suffer from data sparsity. In recent years, many researchers have proposed different methods to alleviate the sparsity, such as recommendation methods based on imputation data (Wen, Z. et al., 2022; Li, Ye. et al., 2019; Hwang, W. S. et al., 2018; Suganeshwar, G. et al., 2020) and recommendation methods based on social networks (Ojagh, S. et al.,2020; Wei, M. S. et al.,2021; Yu, W. et al., 2018), etc. At present, recommendation methods based on singular value decomposition(SVD) have been widely used (Cui, L. Z. et al., 2018).

The above methods alleviate the sparsity problem, but the recommendation quality is still not high (Chai, Z.Y. et al., 2019). The recommendation method based on singular value decomposition also show this deficiency. How to effectively fill the data of singular value decomposition method is a problem worth studying (Chen, V. X., & Tang, T. Y., 2019). To solve this problem, this paper proposes an improved Tanimoto similarity coefficient calculation method to calculate the target user's neighbor users. It is noted that the original Tanimoto similarity coefficient calculation method does not consider user preferences, but only considers whether users have common scoring data. The improved calculation method takes user preferences into account, which can detect similar users in a more reasonable way. Based on the scoring data of the neighbors, the effective estimated data is generated, and finally the original data and the generated estimated data are all used for SVD model training. This method effectively alleviates the data sparsity problem of the method based on singular value decomposition and improves the recommendation quality.

2 RELATED WORK

2.1 Collaborative Filtering Recommendation Method

CF algorithm is divided into the model-based and the neighbor-based. The model-based CF refers to the SVD method (Al-Sabaawi, A. M. A., Karacan, H., & Yenice, Y. E., 2021). The neighbor-based refers to generating a recommendation list for a user based on the preferences of nearby users. Various similarity measurement techniques are used in the CF algorithm to calculate the similarity between items and between users. Most of these methods use co-scoring to calculate the similarity. One of the similarity measurement methods is Tanimoto similarity, which ignores the absolute value of the score and the average score of the user (Zhang, Qin. et al., 2019), and uses the ratio of the intersection and the union of the number of scores to measure the similarity. The neighbor-based CF predicts the score of the target object through the scores of other objects similar to the target object (Yuan, X. F. et al., 2019), which is a commonly used method for predicting missing values. The neighbor-based CF first calculates the similar users of the target user through the traditional similarity calculation formula, and then performs the score prediction of missing values.

2.2 Traditional Similarity Calculation Method

The recommendation system recommends items for target users based on similar user behaviors. How to find effective similar users to target users has always been the focus of researchers. The selection of neighbors is based on the results calculated by the similarity calculation method; thus,

the identification of effective similar users is essential to find an effective method for calculating similarity. Several common similarity calculation methods are as follows:

1) Cosine similarity:This method uses the angle between two vectors to indicate similarity. The specific calculation method is as follows:

$$sim(a,b) = \cos(a,b) = \frac{\boldsymbol{a} \bullet \boldsymbol{b}}{|\boldsymbol{a}||\boldsymbol{b}|} \tag{1}$$

Where a and b respectively represent two different users, and **a** and **b** are the vectors of users a and b in the multidimensional item space.

2) Modified cosine similarity:The modified cosine similarity takes into account the user's rating preferences, and the calculation method is more reasonable. The specific calculation method is as follows:

$$sim(a,b) = \frac{\sum_{i \in I_{ab}} (R_{ai} - \overline{R_a})(R_{bi} - \overline{R_b})}{\sqrt{\sum_{i \in I_a} (R_{ai} - \overline{R_a})^2} \sqrt{\sum_{i \in I_b} (R_{bi} - \overline{R_b})^2}} \tag{2}$$

R_{ai} and R_{bi} represent the ratings of users a and b on item i respectively; $\overline{R}_a$ and $\overline{R}_b$ represent the average ratings of users a and b respectively; I_{ab} represents the items that users a and b have rated together; I_a and I_b are composed of items that users a and b have rated.

3) Pearson correlation coefficient: This is also a commonly used method for calculating similarity,and the value range is [-1,1]. The larger the value is, the stronger the correlation will be. The specific calculation method is as follows:

$$sim(a,b) = \frac{\sum_{i \in I_{ab}} (R_{ai} - \overline{R_a})(R_{bi} - \overline{R_b})}{\sqrt{\sum_{i \in I_{ab}} (R_{ai} - \overline{R_a})^2} \sqrt{\sum_{i \in I_{ab}} (R_{bi} - \overline{R_b})^2}} \tag{3}$$

There are two defects in traditional similarity calculation:(1)Overestimation of the user similarity (Chen, Y. et al., 2021), meaning that the active users in the similarity calculation will be similar to many users, but the real situation is that these users are not very similar to the active users. It is just that the rating data of these users is scarce, and it is difficult to find effective similar users of these users.(2)The target user lacks similar users (Amer, A .A. et al., 2021), which means that when the target user does not have enough similar users, some users who are not similar to the user will be included when selecting the top N similar users for each user.

2.3 Singular Value Decomposition Method

The SVD method in matrix factorization is often used in recommendation systems. Singular value decomposition refers to mapping both users and items onto an f-dimensional space that is not directly observed and is often called a hidden factor (He, Jing., & Hu, Jie., 2021). The calculation formula of this method is as follows:

$$\boldsymbol{R} \approx \boldsymbol{PQ} \tag{4}$$

where $\boldsymbol{R}$ is the user-item rating matrix, and $\boldsymbol{P}$ and $\boldsymbol{Q}$ are two low-rank matrices after decomposition. That is, choose an f smaller than m and n to estimate the matrix $\boldsymbol{P}$ with dimension n×f and the matrix $\boldsymbol{Q}$ with dimension f×m, and f is the hidden factor. User u's predicted score $\hat{r}_{ui}$ for item i can be expressed as qT ip_u, and q_i and p_u are the i-th column and the u-th row of matrix $\boldsymbol{Q}$ and matrix $\boldsymbol{P}$. Due to the different preferences of each user, some users prefer to give high scores, and some users like to give low scores, and there are also differences between projects. Taking these individual deviations into account, the predicted score $\hat{r}_{ui}$ can be updated as:

$$\widehat{r_{ui}} = \mu + b_i + b_u + q_i^T p_u \tag{5}$$

where μ is the global average score; b_i and b_u are item bias and user bias. By minimizing the loss function, matrix $\boldsymbol{P}$ and matrix $\boldsymbol{Q}$ can be obtained. The loss function is as follows:

$$\min_{p^*,q^*,b^*} \sum (r_{ui} - \mu - b_u - b_i - q_i^T p_u)^2 + \lambda(\|p_u\|^2 + \|q_i\|^2 + b_u^2 + b_i^2) \tag{6}$$

According to the gradient descent algorithm, let the update step size be γ,the recursive formula is obtained as:

$$\begin{aligned} p_u &\to p_u + \gamma\left(\left(r_{ui} - \hat{r}_{ui}\right) q_i - \lambda p_u\right) \\ q_i &\to q_i + \gamma\left(\left(r_{ui} - \hat{r}_{ui}\right) p_u - \lambda q_i\right) \\ b_u &\to b_u + \gamma\left(\left(r_{ui} - \hat{r}_{ui}\right) - \lambda b_u\right)\right) \\ b_i &\to b_i + \gamma\left(\left(r_{ui} - \hat{r}_{ui}\right) - \lambda b_i\right)\right) \end{aligned} \tag{7}$$

The regularization parameter λ and the update step size γ are obtained by minimizing the loss function.

2.4 Imputation Data Method

Fill data is mainly used to alleviate the problem of data sparsity. The commonly used fill data are mean fill and mixed fill. (Lee, S. E., Ihm, S. Y., & Park, Y. H., 2021).

Ranjbar et al. (Ranjbar, M. et al., 2015) proposed a method using pre-estimated scoring data to alleviate the problem based on non-negative matrix decomposition. Compared with the original NNMF, it has a lower error. Ma et al. (Hao, M., Irwin, K., & Lyu, M. R., 2007) proposed a neighborhood-based collaborative filtering imputation data method called EMDP. This method can better find similar users by adding a parameter, and then uses an efficient algorithm to predict missing values. The experimental results show that this method is superior. Tripathi, M et al. (Tripathy, M. et al.,

2022) used the method of mixed imputation data to fill the matrix, and combined with clustering method to generate recommendations for users with better results. Yuan, XF et al. (Yuan, XF et al., 2021) proposed a factorization method based on raw data, which generates imputed data through neighbors. The experimental results are better than the existing methods.

The above method cannot fundamentally solve the sparsity problem, and the recommendation accuracy is not high. This problem also exists in the SVD model (Cui, Z.H., et al., 2021), For the SVD model, the more the training data is, the higher the model accuracy will be. This paper proposes an effective interpolation method to improve the recommendation accuracy based on the singular value decomposition model.

3 SINGULAR VALUE DECOMPOSITION ALGORITHM BASED ON DATA FILLING

The Tanimoto similarity coefficient calculation method ignores the absolute value of user ratings when calculating the similarity between users. In order to solve this problem, this paper proposes an improved Tanimoto similarity calculation formula, which fully considers the absolute value of user ratings, obtains more reasonable neighbor users, and then generates effective filling data. Finally, the generated filling data and the original data is both used for SVD model training. The details are as follows.

3.1 Tanimoto Similarity Coefficient

The Tanimoto similarity coefficient is a method of calculating user similarity, which uses common rating items between users and ignores the absolute value of ratings,the Tanimoto similarity coefficient calculation method can calculate the similarity between users under the condition of very sparse data, which is defined as follows:

$$T(a,b) = \frac{\left|I_a \cap I_b\right|}{\left|I_a \cup I_b\right|} \tag{8}$$

I_a and I_b represent the items rated by user a and the items rated by user b respectively.

According to the calculation formula of the Tanimoto similarity coefficient, when the items rated by two different users are completely consistent, the Tanimoto similarity coefficient takes a value of 1, and when the two users have no common rated items, the Tanimoto similarity coefficient takes a value of 0. The closer the result is to 1, the more similar the two users are, and the greater the result value is.

The Tanimoto's similarity coefficient calculation method only considers the number of common ratings, ignoring the problem of the rating value, so it is difficult to effectively find similar users of the target user and needs to be improved.

3.2 Modified Tanimoto Similarity Coefficient

The disadvantage of Tanimoto's similarity coefficient is that it only considers the number of items that are jointly rated and does not consider the difference in the value of the score. Therefore, the improved similarity calculation formula proposed in this paper takes into account the difference in the score value, and is able to find the neighbor users of the target user more reasonably.

Through the above analysis, if two users have rated the same item and the difference between the ratings is less than a certain range, it can be considered that the two users have the same preference for the item. In this paper, a threshold function is set to judge the preferences of different users for the same item. If the difference between the ratings of two users a and b on an item i is less than the threshold δ, it is considered that they have the same interest in this item; otherwise, their interest is considered different. The formula is as follows:

$$N(a,b) = \begin{cases} N(a,b)+1 & \left|R_{aj} - R_{bj}\right| \le \delta \\ N(a,b) & \left|R_{aj} - R_{bj}\right| > \delta \end{cases} \tag{9}$$

R_{aj} and R_{bj} represent the scores of user a and user b on the common scoring item j respectively, the selection of the threshold δ is determined according to the scoring rules, and N(a,b) is initialized to 0.

The improved Tanimoto similarity coefficient is defined as follows:

$$T\_(a,b) = \frac{\left|N(a,b)\right|}{\left|I_a \cap I_b\right|} \tag{10}$$

The improved Tanimoto similarity coefficient calculation method, the denominator takes the intersection of two users, and the numerator is a threshold function N(a,b) with an initial value of 0. $|R_{aj}-R_{bj}|$ indicates the absolute value of the rating difference between two users a and b on the same item i. If the value is less than the threshold δ, it is considered that users a and b have the same preference for the item j, then the N(a, b) value plus 1.

Assume that there are the following user-item scoring tables and scoring rules, (see Table 1 and Table 2), and see Table 3 for the comparison of similarity calculation results before and after the improvement.

In Table 1, the similarity value of user a and user b before the correction is the same as that of user a and user b, both of which are 1 and the similarity score of user a and user d is 0.75. From the scoring rules, user a and user c are the most similar, and the score should be close to 1. It is not difficult to find that the interests of user a and user b are basically opposite, i.e., the similarity calculation result

Table 1.
User-Item rating data

Item	1	2	3	4
User a	2	1	2	5
User b	5	5	5	1
User c	3	1	2	5
User d	3	1	0	4

Table 2.
User-Item scoring rules

0	1	2	3	4	5
Not Rated	Very Annoying	Do Not Like	General	Like	Very Like

Table 3.
Comparison of user similarity calculation results before and after Tanimoto similarity coefficient improvement

	User b	User c	User d
User a Before Improvement	1	1	0.75
User a After Improvement	0	1	1

should be as close to 0 as possible. This is why the calculation of the Tanimoto similarity coefficient does not consider the value of the score and the result does not match the expectation. When the threshold δ is set to 1, using the improved Tanimoto similarity calculation method is more in line with the actual situation. Therefore, the improved method can be more reasonable.

3.3 Creationt of Imputed Data

The quality of data filling directly affects the accuracy of training results based on the SVD model. This paper uses the improved Tanimoto similarity calculation method to obtain effective similar users of the target user, and establishes a sorted user similarity matrix based on the calculation results, namely T_{mm}:

$$T_{mm} = \begin{bmatrix} T_{11} & T_{12} & \cdots & T_{1m} \\ T_{21} & T_{22} & \cdots & T_{2m} \\ \vdots & \vdots & & \vdots \\ T_{m1} & T_{m2} & \cdots & T_{mm} \end{bmatrix}$$

Among them, the values in the matrix represent the similarity between the sorted users, that is, T_{11} represents the similarity between user 1 and its most similar user, and T_{mm} represents the similarity between user m and its least similar user.

By using the similarity matrix obtained through the above process, a new prediction method is proposed, shown in formula (11):

$$F_{aj} = \begin{cases} \dfrac{\sum_{b \in T_a} T_{ab} r_{bj}}{\sum_{b \in T_a} T_{ab}} & R_{aj} = 0 \\ r_{aj} & R_{aj} \neq 0 \end{cases} \tag{11}$$

Where T_{ab} represents the similarity between the sorted user a's bth similar user and user a; r_{bj} indicates the rating of item j by users similar to user a bth; T_a is the similar user sequence of user a; F_{aj} indicates the predicted value of user a for item j; R_{aj} is not equal to 0 meaning that user a has rated item j, and at this time, the value of F_{aj} does not need to be predicted and the original rating data is directly used. R_{aj} equals to 0 meaning that user a has not rated item j. At this time, it is necessary to pre-estimate the rating of the item based on the similar user rating information.

3.4 Singular Value Decomposition Model Based on Data Filling

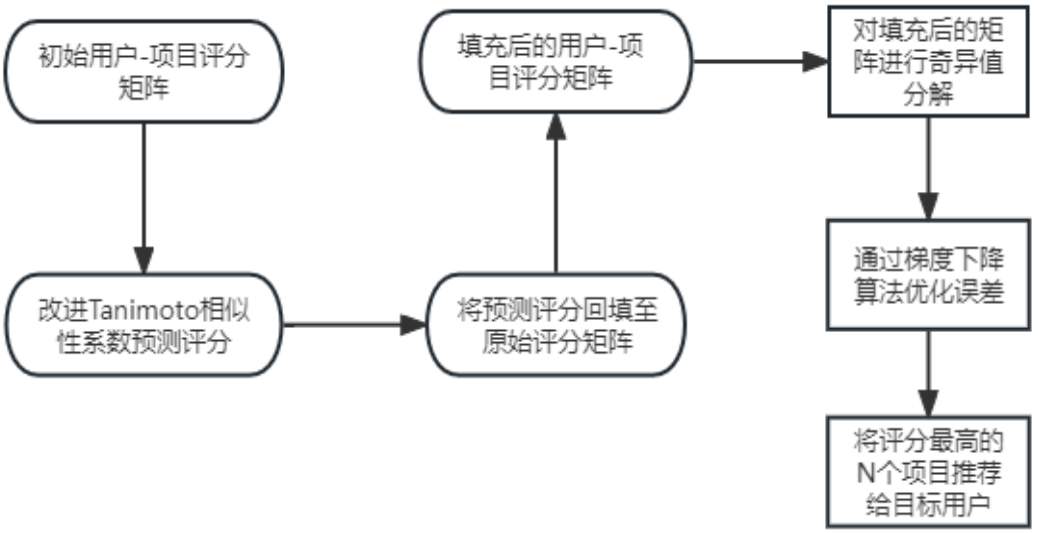
After completing the data filling, this paper puts the filled scoring matrix into the biased SVD model for training, and updates the loss function as:

$$\min_{P,Q,B} \frac{1}{2} \sum_{u=1}^{m} \sum_{i=1}^{n} I_{\varphi}(u,i)(r_{ui} - \widehat{r_{ui}})^2 + \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{n} I_{w}(u,i)(F_{ui} - \widehat{r_{ui}}) + \frac{\lambda}{2}(\|P\|_F^2 + \|Q\|_F^2 + b_u + b_i)^2 \tag{12}$$

Where $\hat{r}_{ui}$ is the predicted value in the SVD model with bias, r_{ui} is the real score value of user u for item i, F_{ui} is the data filling value of user u for item i; $I_{\varphi}(u,i)$ is an indicator function, and if r_{ui} exists, it is 1, otherwise it is 0; $I_w(u,i)$ is also an indicator function, and if F_{ui} exists, it is 1, otherwise it is 0.

3.5 Implementation of the Algorithm

The algorithm proposed in this paper includes three parts: to input scoring data, to calculate the matrix similarity to obtain the neighbor user and predict the score, and to use the filled data and the original data for the SVD model training. First, the collected user behavior data information is converted into the form of user item scoring matrix, and then the user's similar users are obtained according to the improved similarity calculation, and then the data is pre-filled according to the prediction formula, and finally all data are put into the SVD model training. The model diagram of the algorithm is as follows:



Algorithm model in this paper
The specific implementation steps of the SVD algorithm based on data filling are as follows:
1) Use the improved Tanimoto similarity coefficient calculation method to calculate the similarity between all users;
2) Initialize the similarity matrix T_{mm} according to the calculated user similarity;
3) According to the improved similarity method, calculate the estimated data of the target user's unrated items;
4) Reconstruct imputed and raw data into a dense scoring matrix;
5) Singular value decomposition is performed on the dense scoring matrix, the dimension f is specified, and the matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$ are randomly initialized;
6) Update the matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$ according to the stochastic gradient descent algorithm to minimize the loss function;
7) Predict missing values in the original matrix from matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$.

4 EXPERIMENT AND EVALUATION

4.1 Experimental Dataset

This paper uses the MovieLens dataset provided by the GroupLens project team for algorithm verification. The dataset contains 100,000 rating records for 1,682 movies by 943 different users, with ratings ranging from 1 to 5. The level of the score indicates the degree of liking, with 5 being

the most liking and 1 being the least liking. The proportion of ungraded data in this data set reached a very sparse number of 93.7%.

4.2 Metrics

In this paper, two methods of measuring model accuracy: mean absolute error and root mean square error, are used to evaluate the model performance.

1) Mean Absolute Error

The mean absolute error (MAE) is often used to evaluate the accuracy of the model, and the MAE calculation formula is:

$$MAE = \frac{\sum_{i=1}^{N} \left| g_i - h_i \right|}{N} \tag{13}$$

Where g_i is the user's actual score, h_i is the user's predicted score, and N is the number of existing scoring items in the entire project set. The smaller the MAE value is, the smaller the prediction error of the algorithm will be.

2) Root Mean Square Error

Root Mean Square Error (RMSE) is also commonly used to evaluate the model accuracy, and this evaluation index is sensitive to abnormal data. The calculation formula of RMSE is:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N} \left(g_i - h_i \right)^2}{N}} \tag{14}$$

The smaller the RMSE value is, the smaller the prediction error of the algorithm is and the better the performance will be.

4.3 Experimental results and analysis

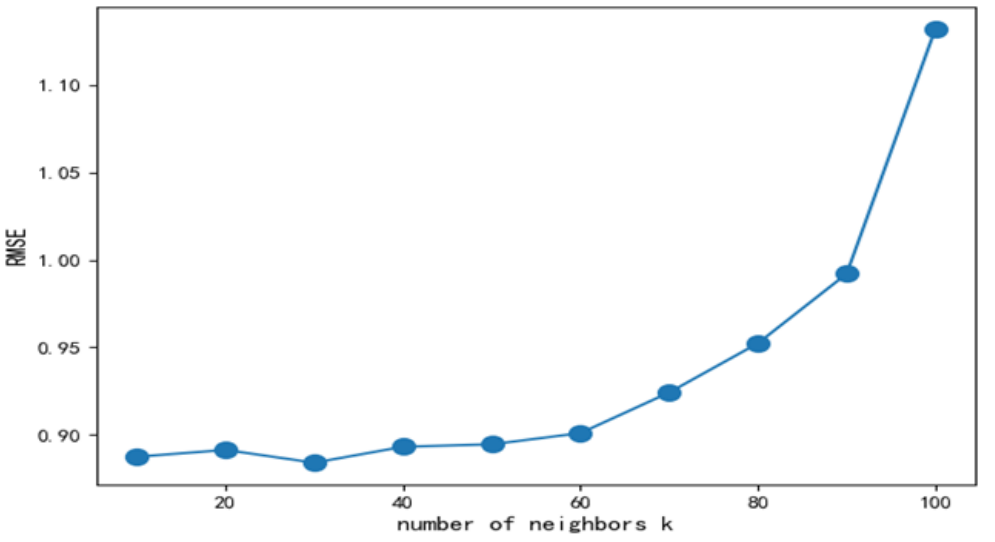
4.3.1 Choice of Parameters in the Model

In this experiment, when calculating the user similarity, the threshold δ is set to 0. In order to improve the recommendation accuracy of the algorithm, this paper analyzes the number k of similar users of the target user in the model, the feature dimension f in the singular value decomposition model, the regularization parameter λ, and the gradient dropping learning rate γ for cross-validation.

1) Selection of parameter k

Figure 1 shows that the number of neighbors starts from 10, and the change of the RMSE value is observed by changing the number of neighbors with a step size of 10. When k=30, the RMSE value is the smallest, i.e., the 30 nearest neighbors of the target user are selected for data filling, and the algorithm performs best at this time.

Figure 1. Changes of RMSE when choosing different neighbor numbers



2) Selection of parameter f

Figure 2 shows that when the hidden factor f is 30, the RMSE value is the smallest, and the algorithm works best.

3) Selection of parameter λ

Figure 2. Changes in RMSE when selecting different values for the feature dimension

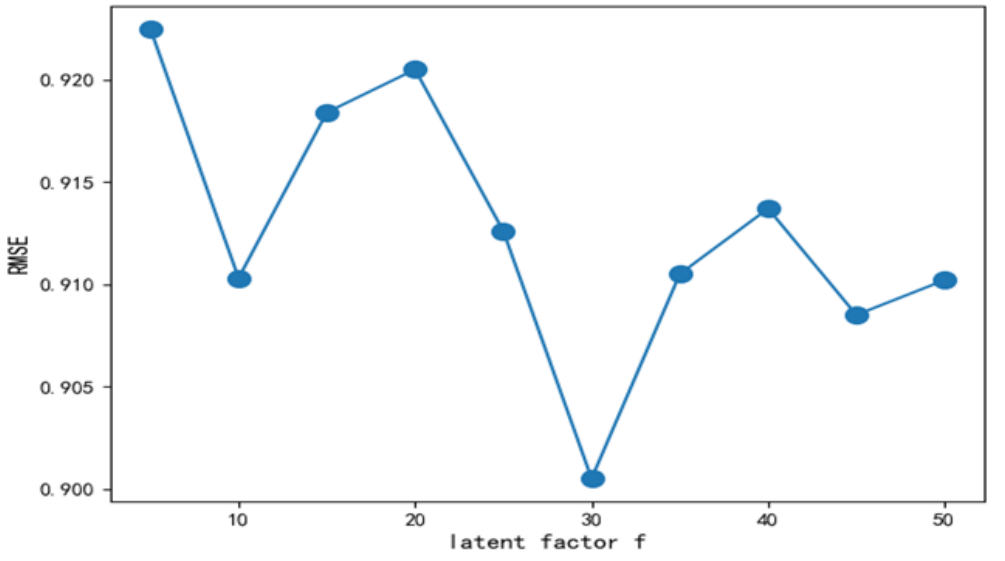


Figure 3. The change of RMSE when the regularization coefficient are at different values

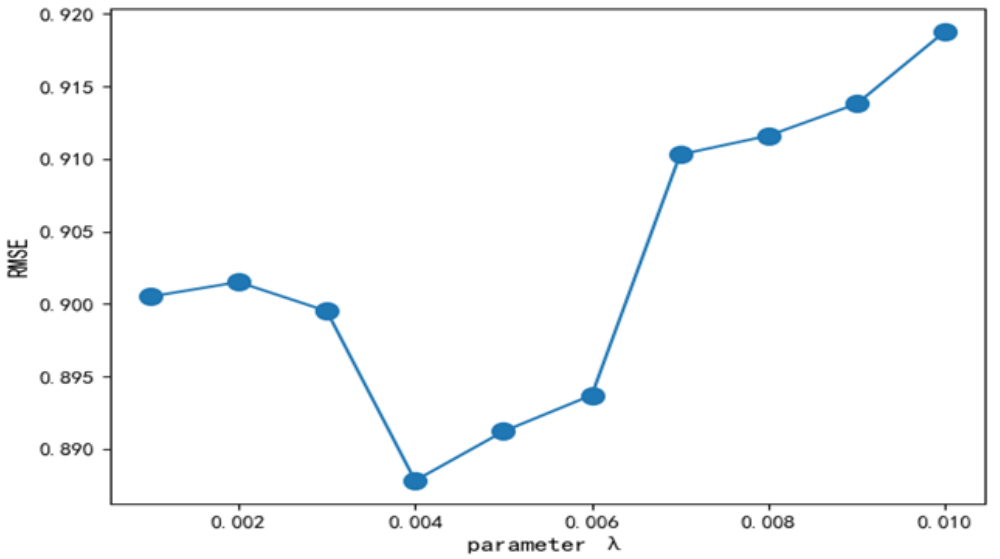
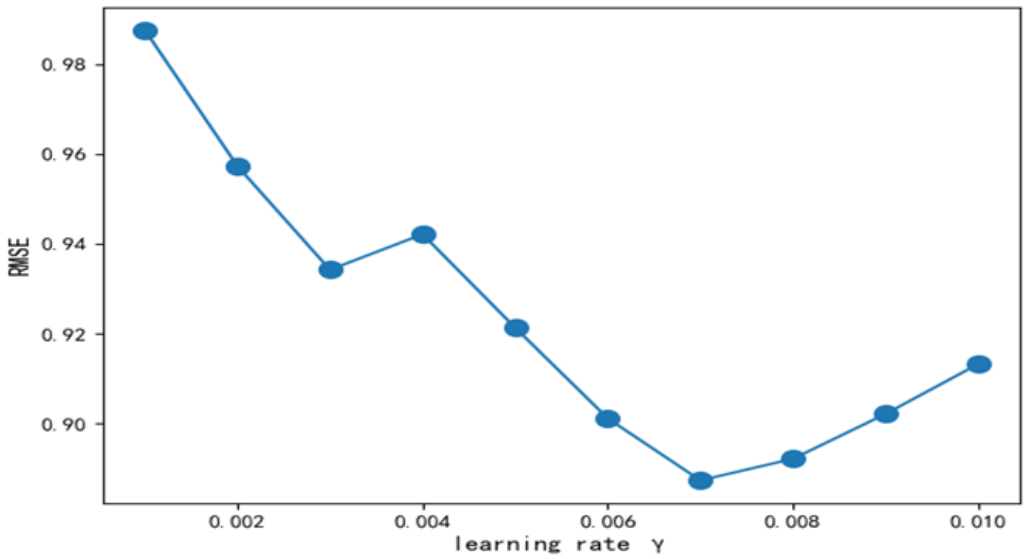


Figure 3 shows that when the regularization coefficient λ is 0.004, the error is the smallest, and the algorithm performs best at this time.

4) Selection of parameter γ

The learning rate γ determines the size of the algorithm in the direction of negative gradient. If γ is too large, the oscillation may occur and the minimum value cannot be found; if γ is too small,

Figure 4. Changes in RMSE when the learning rate is selected at different values



the convergence speed will be too slow, and it may stop at the saddle point. Figure 4 shows that when the learning rate γ is 0.007, the performance of the algorithm is the best.

4.3.2 Result Analysis

The experiment uses the Movielens100K data set, and compares the improved singular value decomposition algorithm with the KNNBasic algorithm, Slope One algorithm, SVD algorithm and SVD++ algorithm. The experimental data is subject to the five-fold cross-validation method, and the experimental results are shown in Figure 5 and Figure 6.

Figure 5. Five-fold cross-validation algorithm MAE comparison

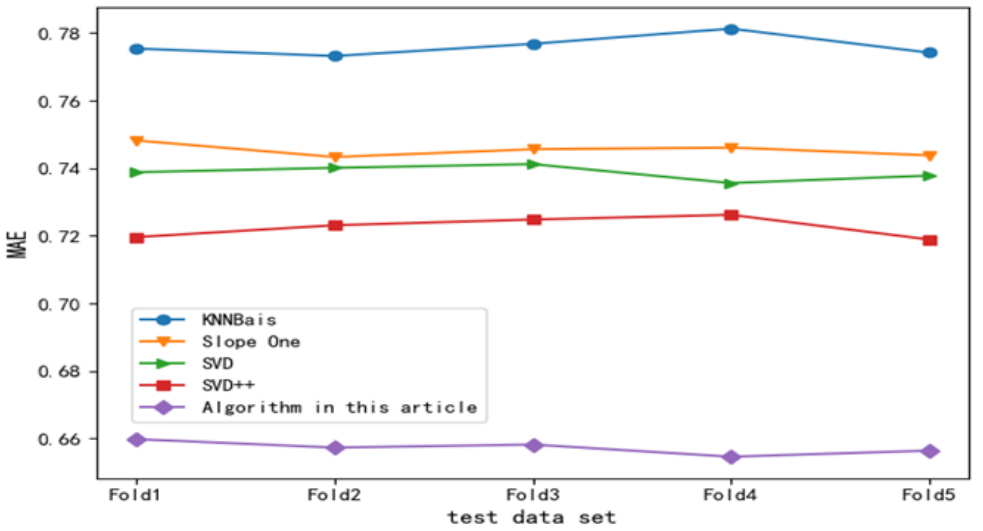
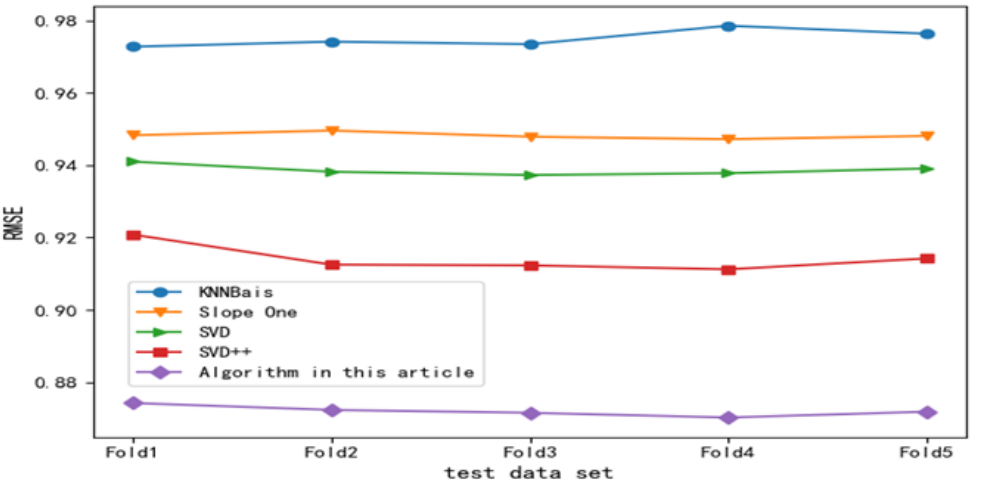


Figure 6. Comparison of RMSE of each algorithm in five-fold cross-validation



The experimental results show that the KNNBasic algorithm has the largest error. The performance of the SVD algorithm is slightly better than that of the Slope One algorithm, and the RMSE and MAE values of the SVD++ algorithm are lower than those of the SVD algorithm. Compared with the other four algorithms, the algorithm proposed in this paper has improved the prediction accuracy of the model. In the best case, RMSE is 10.1% lower than KNNBasic, 7.8% lower than Slope One algorithm, 6.9% lower than SVD algorithm, and 4.8% lower than SVD++ algorithm.

5 CONCLUSION

In order to better solve the problem of data sparsity in the model based on singular value decomposition, this paper proposes a SVD method based on interpolation data. By proposing an improved similarity calculation method and combining the proposed prediction formula, the sparse scoring matrix is filled with data, and finally the filled scoring matrix is trained on the SVD model. The selection of parameters such as feature dimension, regularization coefficient and learning rate in the singular value decomposition model is found by stochastic gradient descent method. The experimental results show that effective filling data for SVD model can have a better recommendation effect. The effective use of additional scoring data is conducive to better data filling. Future work is suggested to work on how to use this information to calculate user similarity and pre-filling.

DATA AVAILABILITY

The data used to support the findings of this study are included within the article.

CONFILICTS OF INTEREST

The author declares that there is no conflict of interest regarding the publication of this paper.

FUNDING STATEMENT

This work was supported by the National Natural Science Foundation of China (No.62262011), The Natural Science Foundation of Guangxi (No.2021JJA170130).

REFERENCES

Al-Sabaawi, A. M. A., Karacan, H., & Yenice, Y. E. (2021). SVD++ and clustering approaches to alleviating the cold-start problem for recommendation systems. *International Journal of Innovative Computing, Information, & Control*, *17*(2), 383–396.

Amer, A. A., Abdalla, H. I., & Nguyen, L. (2021). Enhancing recommendation systems performance using highly-effective similarity measures. *Knowledge-Based Systems*, *217*, 217. doi:10.1016/j.knosys.2021.106842

Chai, Z. Y., Li, Y. L., Han, Y. M., & Zhu, S. F. (2019). Recommendation System Based on Singular Value Decomposition and Multi-Objective Immune Optimization. *IEEE Access: Practical Innovations, Open Solutions*, *7*, 6060–6071. doi:10.1109/ACCESS.2018.2842257

Chen, V. X., & Tang, T. Y. (2019). Incorporating Singular Value Decomposition in User-based Collaborative Filtering Technique for a Movie Recommendation System: A Comparative Study. In *Proceedings of 2019 international conference on pattern recognition and artificial intelligence*. ACM Press. doi:10.1145/3357777.3357782

Chen, Y., Mensah, S., Ma, F., Wang, H., & Jiang, Z. A. (2021). Collaborative filtering grounded on knowledge graphs. *Pattern Recognition Letters*, *151*, 55–61. doi:10.1016/j.patrec.2021.07.022

Chen, Y., & Wang, Y. (2020). Hybrid recommendation algorithm combining content and matrix decomposition. *Jisuanji Yingyong Yanjiu*, *37*(05), 1359–1363.

Cui, L. Z., Huang, W. Y., Yan, Q., Yu, F. R., Wen, Z. K., & Lu, N. (2018). A novel context-aware recommendation algorithm with two-level SVD in social networks. *Future Generation Computer Systems-The International Journal of Escience*, *86*, 1459–1470. doi:10.1016/j.future.2017.07.017

Cui, Z. H., Zhao, P., Hu, Z. M., Cai, X. J., Zhang, W. S., & Chen, J. J. (2021). An improved matrix factorization based model for many-objective optimization recommendation. *Information Sciences*, *579*, 1–14. doi:10.1016/j.ins.2021.07.077

Guo, X., Yin, S. C., Zhang, Y. W., Li, W., & He, Q. (2019). Cold Start Recommendation Based on Attribute-Fused Singular Value Decomposition. *IEEE Access: Practical Innovations, Open Solutions*, *7*, 11349–11359. doi:10.1109/ACCESS.2019.2891544

Hao, M., Irwin, K., & Lyu, M. R. (2007). Effective missing data prediction for collaborative filtering. In *Proceedings of the 30th SIGIR International Conference on Research and Development in Information Retrieval*. ACM Press.

He, J., & Hu, J. (2021). Personalized Recommendation Algorithm Combining Matrix Decomposition and XGBoost. *Journal of Chongqing University*, *44*(01), 78–87.

Hwang, W. S., Li, S. Y., Kim, S. W., & Lee, K. (2018). Data Imputation Using a Trust Network for Recommendation via Matrix Factorization. *Computer Science and Information Systems*, *15*(2), 347–368. doi:10.2298/CSIS170820003H

Jia, J., Liu, P., & Chen, W. (2021). Improved Recommendation Algorithm Based on Matrix Decomposition with Social Information. *Computer Engineering*, *47*(09), 97–105.

Lee, S. E., Ihm, S. Y., & Park, Y. H. (2021). A Technique of Recursive Reliability-Based Missing Data Imputation for Collaborative Filtering. *Applied Sciences-Basel*, *11*(8).

Li, Ye., Wu, C., & Qiang, B. (2019). Multi standard collaborative filtering based on matrix filling and improved PSO algorithm. *Computer Engineering*, *45*(12), 176–181.

Luo, X., Xia, Y., & Zhu, Q. (2012). Incremental collaborative filtering recommender based on regularized matrix factorization. *Knowledge-Based Systems*, *27*, 271–280. doi:10.1016/j.knosys.2011.09.006

Mohammadi, M., Naree, S. A., & Lati, M. (2020). User-item content awareness in matrix factorization based collaborative recommender systems. *Intelligent Data Analysis*, *24*(3), 723–739. doi:10.3233/IDA-194599

Ojagh, S., Malek, M. R., Saeedi, S., & Liang, S. (2020). A location-based orientation-aware recommender system using IoT smart devices and Social Networks. *Future Generation Computer Systems-The International Journal of Escience*, *108*, 97–118. doi:10.1016/j.future.2020.02.041

Ranjbar, M., Moradi, P., Azami, M., & Jalili, M. (2015). An imputation-based matrix factorization method for improving accuracy of collaborative filtering systems. *Engineering Applications of Artificial Intelligence*, *46*, 58–66. doi:10.1016/j.engappai.2015.08.010

Suganeshwar, G., & Ibrahim, S. P. S. (2020). Rule-Based Effective Collaborative Recommendation Using Unfavorable Preference. *IEEE Access: Practical Innovations, Open Solutions*, *8*, 128116–128123. doi:10.1109/ACCESS.2020.3008514

Tripathy, M., Champati, S., & Bhuyan, H.K. (2022). Knowledge Discovery in a Recommender System: The Matrix Factorization Approach. *Journal of Information & Knowledge Management, 21*(04).

Wei, J., He, J. H., Chen, K., Zhou, Y., & Tang, Z. Y. (2017). Collaborative filtering and deep learning based recommendation system for cold start items. *Expert Systems with Applications*, *69*, 29–39. doi:10.1016/j.eswa.2016.09.040

Wei, M. S., Wu, J., Yang, L. N., & Tang, Y. Y. (2021). Matrix factorization with dual-network collaborative embedding for social recommendation. *International Journal of Wavelets, Multresolution, and Information Processing*, *19*(05), 2150016. doi:10.1142/S0219691321500168

Wen, Z., & Shen, S. (2022). Xiangbin, Zhou., Haojie, Lan., Zhiheng, Zhang. Sparse matrix interpolation recommendation technology based on scoring habit weighting. *Jisuanji Yingyong Yanjiu*, *39*(07), 2058–2062.

Yu, W., & Li, S. J. (2018). Recommender systems based on multiple social networks correlation. *Future Generation Computer Systems-The International Journal of Escience*, *87*, 312–327. doi:10.1016/j.future.2018.04.079

Yuan, X. F., Han, L. X., Qian, S. B., Xu, G. X., & Yan, H. (2019). Singular value decomposition based recommendation using imputed data. *Knowledge-Based Systems*, *163*, 485–494. doi:10.1016/j.knosys.2018.09.011