Computing and Informatics, Vol. 42, 2023, 1445–1470, doi: 10.31577/cai_2023_6_1445

MICRO-DIRECTIONAL PROPAGATION METHOD BASED ON USER CLUSTERING

Yuxi Ban, Yuwei Liu

School of Automation
University of Electronic Science and Technology of China
Chengdu 610054, China
e-mail: yuxi.ban@std.uestc.edu.cn, liuyw1993@163.com

Zhengtong Yin

College of Resource and Environment Engineering
Guizhou University
Guiyang 550025, China
e-mail: ztyin@gzu.edu.cn

Xuan Liu

School of Public Affairs and Administration
University of Electronic Science and Technology of China
Chengdu 611731, China
e-mail: liuxuan@uestc.edu.cn

Mingzhe Liu

School of Data Science and Artificial Intelligence
Wenzhou University of Technology
Wenzhou 325000, China
e-mail: liumz@cdut.edu.cn

Lirong Yin*

Department of Geography and Anthropology
Louisiana State University
Baton Rouge, LA 70803, USA
e-mail: lyin5@lsu.edu

Xiaolu Li

School of Geographical Sciences
Southwest University
Chongqing 400715, China
e-mail: `xliswu@swu.edu.cn`

Wenfeng Zheng*

School of Automation
University of Electronic Science and Technology of China
Chengdu 610054, China
e-mail: `winfirms@uestc.edu.cn`

Abstract. With the development of recommendation technology, it is of great significance to analyze users' digital footprints on social networking sites, extract user behavior rules, and make a relatively accurate assessment of each user's needs, to provide personalized services for users. It has been found that the users' behavior on social networking sites has a great correlation with the user's personalities. The OCEAN model's five characteristics can cover all aspects of user personality. There are some shortcomings in the current micro-directional propagation method. This paper has improved the traditional collaborative filtering method and proposed a collaborative filtering method based on user clustering. The model first clusters the users according to their OCEAN model, and then it filters the users collaboratively in the cluster to which the user belongs with the collaborative filtering method based on an optimized singular value decomposition (SVD) recommendation algorithm, called the BiasSVD recommendation algorithm – to reduce the dimensionality of the data. Then it generates recommendations. Experiments show that clustering users' OCEAN models can improve the accuracy of recommendations. Compared with the entire user space, it greatly reduces the recommendation time, and effectively solves the cold start problem in micro directional propagation.

Keywords: OCEAN model, micro-directional, propagation clustering, recommendation algorithm, collaborative filtering, BiasSVD, cold start

* Corresponding author

1 INTRODUCTION

The Internet has now entered every corner of people's daily lives [1, 2]. People can do a lot of things via the internet, such as reading the news, finding jobs, and watching movies. There are certain hidden rules in the various unregulated behaviours of users on the Internet [3, 4, 5].

When users browse the Internet, there will always be a variety of recommendation pages. The micro-directional recommendation can provide users with more suitable choices and save users' browsing time. Therefore, it is of great significance to analyze users' digital footprints on social networking sites and to assess each user's needs based on their personality and style to provide users with micro-directional content dissemination [6].

Although the micro-directional propagation system is constantly becoming more perfect by adding new technologies and algorithms [7, 8], some problems still need to be revised. The main problems at present are as follows.

In micro-directional propagation systems, there is a very common problem of sparsity in the matrix data, which generally reduces the accuracy of micro-directional propagation. Several corresponding methods have been developed to solve the problem of sparsity, the most common one is to fill in a fixed value to make the original matrix a dense matrix [9, 10]. The method of mean padding is generally used, that is, the average value of all users' ratings is obtained, and the matrix is filled with the average value.

However, there are also certain problems with this method. Under normal circumstances, the user's scoring standard is inconsistent, which makes it difficult to deal with the problem in essence. Therefore, after exploring the attributes of users and items, researchers have tried to solve the problem of data sparsity by using other solutions. Solutions include collaborative filtering based on item similarity, dimensionality reduction on sparse matrices, AI technology [11, 12], and so on. Compared with the similarity between users, the similarity between the calculations makes the model more stable. The dimensionality reduction method mainly includes singular value decomposition (SVD), principal component analysis algorithm (PCA) [13], and latent semantic indexing algorithm (LSI).

To make the recommendation more accurate, the system must predict the user's preferences based on the user's prior rating of the item. There are currently several solutions to the cold start problem, one of which is the hybrid recommendation algorithm. It is a model that effectively blends the collaborative filtering algorithm with the content-based recommendation algorithm [14, 15]. New items are often added to the system, but they are usually ignored because the collaborative filtering recommendation algorithm only recommends the one to the users according to their preference, therefore, the properties of the new items cannot be used. Only after it has been evaluated by many users, the new item is likely to be recommended. Therefore, a hybrid micro-directional propagation algorithm can be used.

After research, a model-based micro-directional propagation algorithm is usually used. This algorithm customizes a basic model for users, then estimates the items of interest to them based on their data, and continuously improves the model. Currently, the commonly used methods are clustering, Bayesian Networks [16], dimensionality reduction techniques, and so on. The recommendation idea based on the clustering technique is to classify users firstly through certain attributes and search for their nearest neighbour, which can greatly narrow the search space and improve the real-time performance of the micro-directional communication. The implementation method based on the Bayesian network is to create an estimated model for the user, which is usually more accurate and time-sensitive than the micro-directional propagation result of the traditional models, but the creation process of the model is cumbersome.

Based on the above problems, a new collaborative filtering method based on user OCEAN model clustering is proposed in this paper. The OCEAN model is a theoretical model that can represent personality characteristics. The idea of this model is to find vocabulary that can summarize the user's personality, and then cluster the vocabulary to form a description of the personality. The OCEAN model includes five personality traits: Open, Conscientiousness, Extraversion, Agreeability and Neuroticism. These five factors provide a rich personality structure. And psychologists' research has found a strong connection between the OCEAN theoretical model and users' digital footprints on social media sites [7]. The OCEAN model can effectively predict the behavior of users on social networks, and similarly, the behavior of users on social networks can also reflect their OCEAN model.

At present, there is no micro-directional communication system that can solve the scalability and sparsity problem of the micro-directional communication system by combining the user's personality characteristics with matrix dimension reduction. Therefore, to improve the accuracy of prediction and overcome the sparsity problem of the recommendation system, this study tries to develop a micro-directional content dissemination system based on the clustering of the user OCEAN model.

The work of this paper is as follows:

- Sina Weibo was selected as the main research scenario to obtain experimental data.
- Establish the OCEAN model for Sina users through the OCEAN model test scale.
- Establish a user-item scoring matrix.
- Recommend movies to users by the traditional collaborative filtering method and SVD-based collaborative filtering method respectively.
- Use the BiasSVD model to improve the collaborative filtering system.
- The improved algorithm flowchart is shown in Figure 1.

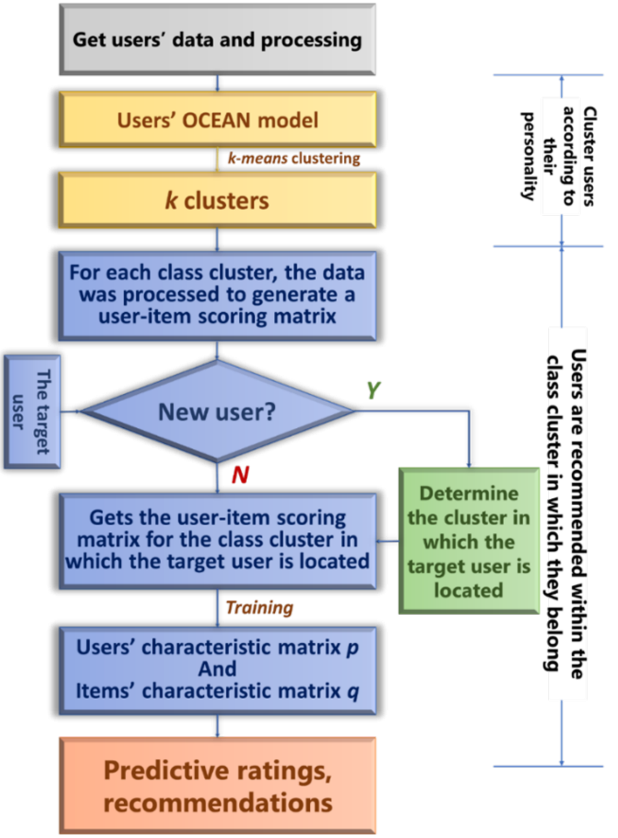


Figure 1. Improved algorithm flowchart

2 DATASET

2.1 Collecting Data

In the micro-directional communication system, it is necessary to collect users' historical data and understand their habits to recommend products to users. There are two types of user behavior data: explicit data and implicit data [17]. Explicit user behavior data refers to the digital footprints that users leave on websites, such as Weibo, movie ratings, etc. [18]. Implicit user behavior data is generated unintentionally in the process of the user to achieve other purposes in daily life, such as the number of times the user has visited the website, page pause time, etc. [19].

To establish a basic dataset, this study selected the most recognized OCEAN model test scale in psychology – the NEO Personality Questionnaire. The NEO personality questionnaire is an OCEAN model questionnaire developed by American psychologists Costa and McRae based on the structure of five aspects of human personality [20].

Sina Weibo is one of the mainstream social media websites at present, with a large number of active users, and its user information is mostly public. Based on the above characteristics, this study chooses Sina Weibo as the main research scenario.

Based on the above description, the NEO questionnaire for testing the OCEAN model was distributed to Sina Weibo users. Specific screening criteria are as follows.

1. The Weibo accounts provided by the tested users are accurate;
2. The time for respondents to answer the questionnaire is longer than 200 seconds;
3. The number of tweets in the corresponding Weibo account of the subject user is more than 100.

To meet the experimental requirements of this paper, a crawler program based on Java was developed, and some data from Sina Weibo users were selected as the data set of this experiment. The selection criteria for Sina Weibo users are:

1. The user's Sina Weibo homepage contains data on movie ratings;
2. The number of microblogs posted by the user is more than 100;
3. The user does not evaluate a movie many times and all the scores are the same;
4. The number of movies evaluated by the user is more than 3.

2.2 Data Processing

According to the above screening criteria, the user OCEAN model was obtained after counting the questionnaire results and processing the data. To more intuitively present the results of the NEO questionnaire, Table 1 shows seven user valid cases.

User ID	Neuroticism (N)	Extraversion (E)	Openness (O)	Agreeableness (A)	Conscientiousness (C)
1	3.167	3.667	3.417	2.917	3.417
2	3.083	2.5	2.333	3.5	2.666
3	2.5	3.167	2.667	3.583	2.333
4	2.5	4.167	3.75	3.5	3.75
5	1.75	2.667	2.917	3.167	4.083
6	2.667	4	3	4	2.667
7	4.417	3.25	2.5	2.5	1.5

Table 1. Valid cases of the user OCEAN model

The scores in Table 1 indicate the bias of the probationary users in the measured feature dimension, and all data are feature dimension values ranging from 0 to 5. The closer the data in the table is to 5, the greater the user's bias in the feature dimension.

After crawling the homepage data of Sina Weibo users who meet the above screening criteria based on the Java program, the film score data of 2012 Sina Weibo users was obtained with a total of 4 579 movies. And after processing, the user-item scoring matrix was got for 2012 Sina Weibo users.

Each user's data is stored in a text document and was processed to get the user's rating of the movie, between 0 and 5. As shown in Table 2.

User ID	Item ID			
	Item 1	Item 2	Item…	Item n
User 1	$r_{1,1}$	$r_{1,2}$	…	$r_{1,n}$
User 2	$r_{2,1}$	$r_{2,2}$	…	$r_{2,n}$
User...	…	…	…	…
User m	$r_{m,1}$	$r_{m,2}$	…	$r_{m,n}$

Table 2. User's score sheet for the item

The data is converted into a matrix form where the row represents the user, the column represents the movie, and the unrated movie is represented by 0, then a matrix is used to represent the user's scoring matrix for the movie:

$$R = (r_{ij}) = \left[\begin{array}{c|ccccc} r_{11} & r_{12} & \dots & r_{1n} & \dots & r_{1N} \\ r_{21} & r_{22} & \dots & r_{2n} & \dots & r_{2N} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ r_{i1} & r_{i2} & \dots & r_{in} & \dots & r_{iN} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ r_{M1} & r_{M2} & \dots & r_{Mn} & \dots & r_{MN} \end{array}\right]. \tag{1}$$

3 APPROACH

3.1 Clustering User Groups Based on OCEAN Model

After pre-processing the data, this study uses k-means clustering method to cluster users according to the user's OCEAN model. K-means clustering method belongs to unsupervised learning. It takes k as a parameter and divides n objects into k clusters so that the similarity between clusters is lower, and the similarity in clusters is higher. In this study, let $\mathrm{x}^{(i)}$ be a five-dimensional vector representing the OCEAN model of one Sina Weibo user. Then the training sample is given as $\mathrm{x}^{(1)}, \mathrm{x}^{(2)}, \dots, \mathrm{x}^{(m)}$ in the clustering algorithm, representing the OCEAN model of the m Sina Weibo users. The specific process is described as follows:

1. Select k users as the cluster centroids $\mathrm{u}_1, \mathrm{u}_2, \dots, \mathrm{u}_k$.
2. Repeat the following process until convergence:

   - Calculate the class cluster to which each sample i should belong:

$$c^{(i)} := \arg\min_j \left\| x^{(i)} - \mu^{(j)} \right\|^2. \tag{2}$$

- Recalculate the centroid of each cluster and update the original cluster center:
$$\mu_j = \frac{\sum_{i=1}^{m} \{c^i = j\} x^i}{\sum_{i=1}^{m} \{c^i = j\}}. \tag{3}$$

The condition for the end of the above algorithm is convergence. The following describes the convergence of the clustering algorithm. Here, the distortion function is defined as Equation (4):

$$J(c,\mu) = \sum_{i=1}^{m} \parallel x^{(i)} - \mu_c^{(i)} \parallel^2. \tag{4}$$

The J function indicates the sum of the squares of the differences between each training sample and the cluster center. Convergence means that the function value is minimized. Assuming that the J function has not yet reached the minimum value, then adjust the class cluster to which each training sample belongs to make the value of the J function smaller, or adjust the centroid of each cluster can also reduce the value of the J function. When the J function is decremented to a minimum, that is, when the centroid is no longer changing, it is considered that the J function is convergent.

In all user spaces, this paper makes clusters according to the users' OCEAN model. After the k clusters are created, the data of the cluster center is stored. When the target user is a new one, it only needs to determine which OCEAN model of the target user is closest to the cluster center, so that the target user is classified into the cluster closest to him. When the target user already exists in the cluster, the data of all the users in the cluster in which the target user is located is used to perform the score prediction and recommendation of the target user.

After considering the personality characteristics of users, this algorithm only needs to calculate the similarity between users and users within the cluster, and cluster users with similar personalities into a cluster. For users with similar personalities, increasing the proportion of ratings for movies of the same type can alleviate the sparsity of the user rating matrix and improve the accuracy of finding nearest neighbors. Therefore, compared to searching the entire user space, the range of search required is greatly reduced, and the clusters to be analyzed are reduced, which means the number of users is reduced. And it greatly improves the calculation speed of the system, reduces the calculation pressure of the system, and ensures the quality of recommendations.

3.2 Collaborative Filtering Based on SVD

Since the user only scores several items, there would be a huge number of items that all users in the system have scored. There is a problem of sparsity in the system of collaborative filtering. To remove noise data from large sparse data sets, the researchers proposed methods of dimensionality reduction. This paper uses singular value decomposition (SVD) [21], which can be seen as a method for the dimensionality reduction of data, and a powerful computational tool for solving data analysis problems in numerical linear algebra.

SVD can decompose any matrix singular value. Using SVD, a matrix $R \in R^{N\times M}$ of rank $k <= min(N, M)$ can be decomposed into:

$$R = U \sum V^T \tag{5}$$

among them:

$$U = [\underbrace{u_1, u_2, \ldots, u_r}_{U_r}, \underbrace{u_{r+1}, u_{r+2}, \ldots, u_N}_{U_r}] = [U_r \mid U_r], \tag{6}$$

$$V = [\underbrace{v_1, v_2, \ldots, v_r}_{V_r}, \underbrace{v_{r+1}, v_{r+2}, \ldots, v_N}_{V_r}] = [V_r \mid V_r], \tag{7}$$

$$R = [U_r \mid U_r] \begin{bmatrix} \Sigma_r & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_r^T \\ V_r^T \end{bmatrix} = U_r \Sigma_r. \tag{8}$$

U is composed of standard orthogonal bases, $\sum$ is a diagonal matrix composed of singular values, and V is composed of another set of standard orthogonal bases.

In most cases, the matrix can be approximated by the first k singular values to achieve the purpose of dimensionality reduction, as shown in Equation (9):

$$R_{m\times n} \approx U_{m\times k} S_{k\times k} V^T_{n\times k}. \tag{9}$$

R is the user's scoring matrix for the item, U is the user concept matrix, its column is the eigenvector of the left singular value, and S is the singular value, which is a diagonal matrix in which the elements are arranged in descending order. V^T represents the item concept matrix, and its rows are feature vectors of right singular values.

The value of k is calculated by formula (10):

$$\frac{\sum_{i=1}^{k} \sigma_i^2}{\sum_{i=1}^{m} \sigma_i^2} \geq \text{percentage}. \tag{10}$$

The 'percentage' represents the threshold of the square of the singular value and the specific gravity. The percentage is less than 1, and the value of the percentage is proportional to the value of k. k is much smaller than m and n in the matrix, thus reducing the dimension of the matrix.

Figure 2 shows the general steps for dimension reduction of user items in collaborative filtering.

Therefore, using the SVD algorithm, a given matrix R can be converted to $R = U \sum V^T$.

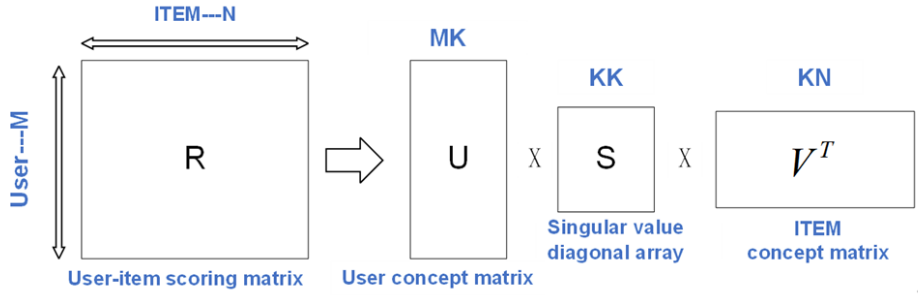


Figure 2. General steps for user matrix decomposition in collaborative filtering

The collaborative filtering algorithm based on the SVD algorithm is divided into two types [22]: user-based SVD micro-directional propagation algorithm and item-based SVD micro-directional propagation algorithm. In the item concept matrix V^T, each column represents an item vector. The greater the similarity between columns, the higher the similarity between the two items. Assuming that user x is to be scored for item y, the formula for the algorithm is as follows:

$$R_{x,y} = U_k \cdot \sqrt{S_k}(x) \cdot \sqrt{S_k} \cdot V_k^T(y). \tag{11}$$

The overall idea of the algorithm is: first find items that have not been scored by users, and then use SVD to compress the user-score matrix to the low-dimensional space to obtain the users' concept matrix and the items' concept matrix. Use the two matrices to obtain the item prediction scores provided by the user, and then the prediction scores were ranked from high to low, and finally return the topN score and recommend it to the user.

There are four steps:

1. To load the datasets;
2. To determine the dimension that should drop by calculating the value of percentage, that is, the value of k;
3. To estimate the missing items of users' scores in the reduced data based on SVD and return the estimated value.

3.3 Collaborative Filtering Base on BiasSVD

The SVD algorithm is considered one of the most commonly used algorithms in the recommendation system, but it has shortcomings [23]:

1. The required matrix of SVD decomposition is the dense matrix, but the actual user-item scoring matrix is very sparse. It takes up huge memory space to complete a large sparse matrix into a dense matrix.

2. In addition, the sparse matrix is usually converted into a dense matrix through completion. In most cases, the average score is used, which would make the errors occur due to the different users or items.
3. Decomposing a matrix into three matrices using the SVD method is time-consuming, which greatly reduces the efficiency and real-time performance of the recommendation system.

Therefore, when faced with the computational efficiency of traditional SVD, the model of BiasSVD was proposed.

It is known that for any matrix, there is an approximate matrix R^f with an optimal rank of f, which can be expressed as the product of two matrices $P_{m\times f}$ and $Q_{f\times n}$.

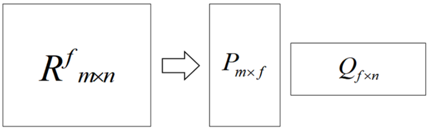


Figure 3. Decomposition of matrix approximation into the product of two matrices

Then there is the same decomposition in the user-item scoring matrix. The decomposition of this matrix is represented by the product of matrix P and matrix Q:

$$R_{m\times n} = P_{m\times f} \bullet Q_{f\times n}, \quad f < m, n. \tag{12}$$

Assume that the known score is represented by r_{ij} and the missing score is represented by $\hat{r}_{ij}$. Therefore, unknown scores can be calculated using formula (13):

$$\hat{r}_{ui} = p_u \bullet q_i. \tag{13}$$

Among them, p_u is the element corresponding to the u^{th} user in the user score matrix, and q_i is the element corresponding to the i^{th} item in the project score matrix.

In the usual scoring system, there are usually three aspects of biasing factors:

1. Some factors that are not related to users or items;
2. The user has some factors that are not related to the item. For example, some users have low scores for all items, this factor is called user bias;
3. The item contains some user-independent scoring factors. If the quality of the item itself is not good, it will directly lead to the user's low score. This factor is called item bias.

BiasSVD is an improved version of the recommendation algorithm for optimizing the SVD method based on the above three bias factors [24]. After adding the bias, the prediction formula for the missing item score is:

$$\hat{r}_{ui} = u + b_u + b_i + p_u \bullet q_i. \tag{14}$$

The average score of all user-item scoring matrices is u, the user deviation item of the i^{th} user is b_i, and the item deviation item of the u^{th} item is b_u. When the matrix is decomposed, the product of P and Q plus the bias item is fitted to the value in the user-item scoring matrix by training. Using the mean square error as the cost function, the formula for optimization is:

$$J = \frac{1}{2}\sum_{u,i}\left(r_{ui} - \mu - b_u - b_i - p_u \cdot q_i\right)^2. \tag{15}$$

To prevent overfitting, add a regularization item to the loss function, so the formula for the objective function is:

$$J = \frac{1}{2}\sum_{u,i}\left(r_{ui} - \mu - b_u - b_i - p_u \cdot q_i\right)^2 + \frac{1}{2}\lambda\left({b_u}^2 + {b_i}^2 + \| q_i \|^2 + \| p_u \|^2\right). \tag{16}$$

The parameters are optimized by a gradient descent method. Finding the partial derivative of each parameter,

$$\begin{aligned} e_{ui} &= r_{ui} - \mu - b_u - b_i - p_u \cdot q_i \\ &\Rightarrow \begin{cases} \frac{\partial J}{\partial b_u} = -e_{ui} + \lambda b_u, \\ \frac{\partial J}{\partial b_i} = -e_{ui} + \lambda b_i, \\ \frac{\partial J}{\partial p_u} = -e_{ui} q_i + \lambda p_u, \\ \frac{\partial J}{\partial q_i} = -e_{ui} p_u + \lambda q_i. \end{cases} \end{aligned} \tag{17}$$

The updated formula for each parameter is:

$$\begin{cases} b_u := b_u + \eta(e_{ui} - \lambda b_u), \\ b_i := b_i + \eta(e_{ui} - \lambda b_i), \\ p_u := p_u + \eta(e_{ui} q_i - \lambda p_u), \\ q_i := q_i + \eta(e_{ui} p_u - \lambda q_i). \end{cases} \tag{18}$$

The p, q, and other parameters can be obtained through several iterations, thereby performing micro-directional propagation to the target user.

4 EXPERIMENT

The experiment was divided into two parts. The first part is the SVD algorithm based on the user OCEAN model, and the second part is the BiasSVD algorithm based on the user OCEAN model.

4.1 SVD Algorithm Based on User OCEAN Model

To test the effectiveness of the algorithm improvement, the entire user space and 20 % of the movie score data of each cluster were respectively set to 0, by comparing the difference between the predicted value of the test data and the actual score value. Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and model running time were used as the evaluation indicators [25, 26]. The verification of the algorithm is completed by comparing the results of this experiment with the results of collaborative filtering without adding the OCEAN model. Validated indicators include the sparsity of the matrix, the running time of the recommendation algorithm, RMSE, and MAE [27].

Before the prediction experiment, a parameter needs to be determined first: percentage (the threshold for the proportion of the sum of squares of the singular values). In the process of finding the best parameters, the value of RMSE is used as a measure.

The experiment process was divided into three steps:

1. Clustering of users based on the OCEAN model;
2. The RMSE, MAE, sparsity, and running time of the recommendation algorithm were calculated in the entire user space;
3. The RMSE, MAE, sparsity, and running time of the recommendation algorithm were computed in each class cluster separately.

The cluster center and the number of centroids need to be determined before the experiment. In this experiment, some microblog users with high traffic and different tags were manually designated as cluster centers, and the number of cluster centers n was assumed to be 6.

Here are the specific steps of the experiment:

1. Determine the optimal percentage value.

   Before making a prediction, the optimal 'percentage' value is determined firstly, that is, the ratio of the reserved features. The 'percentage' and k are proportional. The value of the 'percentage' can be used to calculate the specific value of the dimension k to which the matrix is lowered. The formula is as follows:

$$\frac{\sum_{i=1}^{k} \sigma_i^2}{\sum_{i=1}^{m} \sigma_i^2} \geq \text{percentage}. \tag{19}$$

   In this experiment, the overall user group and each cluster were tested by different 'percentage' values. The experimental results show the RMSE values corresponding to each 'percentage' after matrix decomposition. All users' represents the result analysis of the recommendation in the entire user space without considering the user OCEAN model. "Class cluster i" represents the result of recommending the user within the cluster after clustering by the OCEAN model.

Each cluster and the entire user space are compared in the coordinate system, as shown in Figure 4.

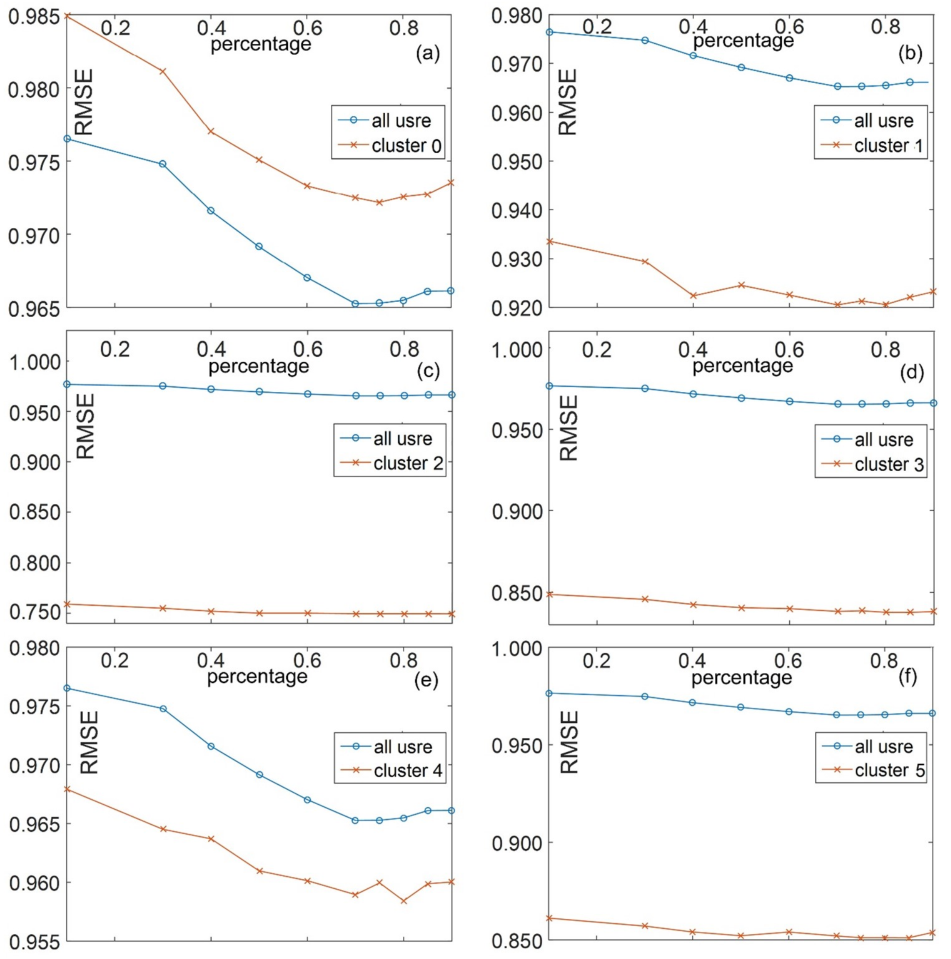


Figure 4. Different percentage values correspond to the experimental results

As can be seen from the results in Figure 4, the RMSE in class cluster 0 is about 0.07, higher than the RMSE of the entire user space, and the RMSE in other clusters is smaller than the RMSE of the long user space. Furthermore, the optimal RMSE averaging for all clusters is 0.88155, which is less than the optimal RMSE value of 0.96526 for the entire user space, demonstrating the effectiveness of adding the user OCEAN model to the recommendation process.

Meanwhile, as the 'percentage' value increases, the RMSE value shows a slight increase after the first drop. This is because, in general, retaining about 80 % of the singular value can cover all the features of the matrix. When the proportion of feature retention is too small, all the features cannot be extracted by singular value decomposition, resulting in low accuracy of recommendation; when the feature retention ratio is too large, the matrix information contains too many hidden features, which affects recommendation accuracy. Therefore, according to the experimental results, the best "percentage" value is between 0.75 and 0.85. In the subsequent experiments, use 'percentage' = 0.8.

2. Comparison of SVDs incorporating OCEAN models with SVDs without OCEAN models.

   This experiment uses the best parameter obtained from the above experimental results to analyze the 'percentage' = 0.8. The SVD-based collaborative filtering results are compared with traditional item-based collaborative filtering (CF) results in the entire user space and each cluster.

4.2 BiasSVD Algorithm Based on User OCEAN Model

To test the effectiveness of improvement, each user-space was divided into the training set and test set respectively, with the proportion of the training set at 80 % and the test set at 20 %. By calculating the difference between the predicted value and the actual score value of the test data in the test set, the indexes (RMSE and MAE) of measuring the accuracy of the prediction were obtained. The results of this experiment were compared with those of BiasSVD without the OCEAN model to verify the improvement. The following are the specific steps of the experiment:

1. To determine the rank after matrix decomposition.

   Before the prediction of item scoring, the rank after matrix decomposition, namely the dimension f of the eigenvector, should be determined first. When calculating the optimal value of the eigenvector f, experiments were conducted on the data with and without regularization items, respectively. In this experiment, RMSE was used as an indicator to measure the accuracy of results.

   The experiment was divided into two groups. One group is the RMSE value of all user space without considering the user OCEAN model. The above results are compared and shown in the coordinate system, as shown in Figure 5.

   The other group is the RMSE value of users who are clustered based on the user OCEAN model. The average value of all cluster results is taken as the experimental results of the second group. As the number of users of each class cluster is different, the formula for calculating the f value is as follows:

$$f = \text{number} \times \text{ratio}. \tag{20}$$

   In the formula, "number" represents the number of users in the class cluster, "ratio" (%) represents a ratio, and the value multiplied by two values represents the value of f. In Figure 6, the value of the ratio is taken as the horizontal

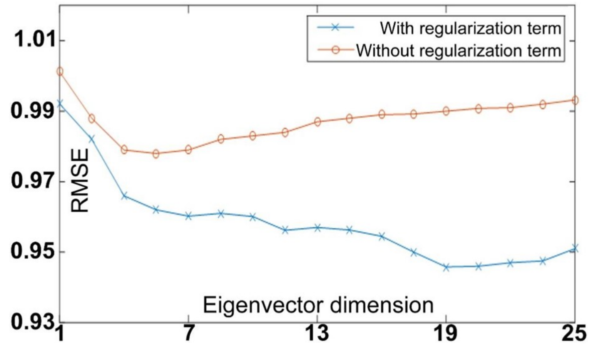


Figure 5. Effect of feature vector dimension f on the entire user space

coordinate, and RMSE's change trend along with the eigenvector dimension is represented on the graph, as shown in Figure 6.

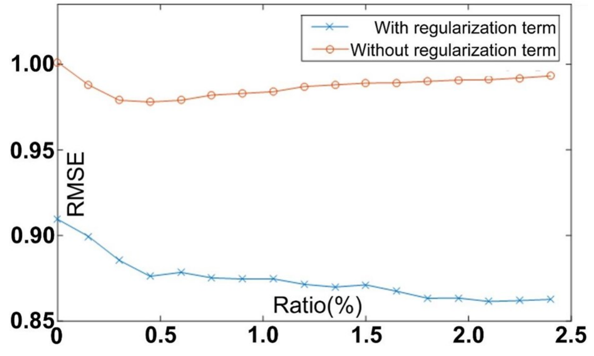


Figure 6. Effect of feature vector dimension on clusters

In general, the more dimensions of the eigenvector set, the more information can be obtained from the data, so the result is more accurate. However, it can be seen from the experimental results in Figures 5 and 6 that the RMSE value decreases with the larger f value regardless of the presence or absence of the regular item. When the regular item is added, the RMSE curve reaches its lowest point when the f value is about 40 (ratio (%) is about 2), and then the curve rises slightly and the model appears the overfitting phenomenon. Without the regular item, the RMSE value reaches its lowest point much earlier.

Therefore, it can be seen that the regular item can avoid the phenomenon of overfitting to a certain extent. Then in the following experiment, the influence of regularization parameters on experimental results will be judged by changing the value of regularization parameters.

It can also be seen from Figure 4 that the RMSE value of the OCEAN model without considering users is between 0.82 and 0.88, while that of the OCEAN model without considering users is between 0.74 and 0.8, which proves the effectiveness of the experimental improvement.

2. To determine the best regularization parameter value for experimental results.

   In this experiment, the user-item scoring matrix obtained from all users' data is used to select regularization parameters in the case that other parameters remain unchanged, and the regularization parameter is increased from 0 to 0.1 to determine the best regularization parameter value for experimental results.

   The changing trend of RMSE with the regularization parameter is shown in the graph, as shown in Figure 7.

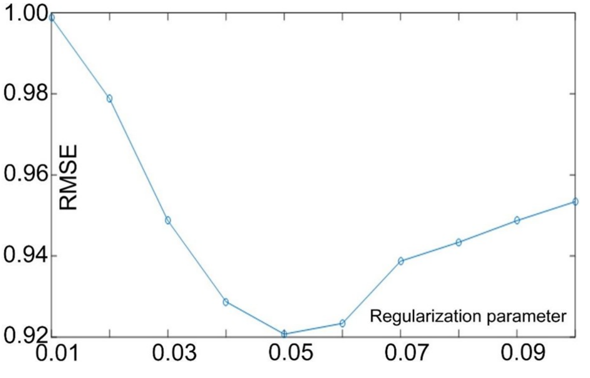


Figure 7. Selection of regularization parameters

According to the curve, the RMSE value of the model first decreases with the increase of regularization parameters within the range of 0-0.04 and then increases with the increase of the regularization parameter. Therefore, when the regularization parameter is too small, the system appears to have the phenomenon of overfitting. When the regularization parameter is large, the system is underfitted. Based on the above experiments, the value of the regularization parameter was set as 0.05 in the following experiments.

4.3 Evaluation

The evaluation index is used to evaluate whether the recommended item in the micro-directional content distribution system is consistent with the item of interest to the user, that is, the index for evaluating the accuracy of the recommendation is high enough, which represents the performance of the system. To judge the accuracy of the prediction of this experiment, root means square error (RMSE) and mean absolute error (MAE) were taken as two of the evaluation indicators.

$$\mathrm{RMSE} = \sqrt{\frac{1}{|R|}\sum_{(i,p)\in R}(P_{i,p} - R_{i,p})^2}, \tag{21}$$

$$\mathrm{MAE} = \frac{1}{|R|}\sum_{(i,p)\in R}|P_{i,p} - R_{i,p}|. \tag{22}$$

$P(i,p)$ represents the score of user p to item i estimated by the micro-directional propagation algorithm, $R(i,p)$ represents the actual user p scoring the item i. R represents the estimated number of items included. A smaller RMSE indicates a higher score accuracy for the prediction. The same as MAE.

The sparser the matrix, the less information is passed in the matrix, which can reduce the accuracy of the scoring of the item. Therefore, the experiment regarded the sparseness of the matrix as one of the evaluation criteria. Under the condition of ensuring prediction accuracy, the high efficiency of the recommendation system is also an important index. Therefore, the running time is also taken as one of the evaluation indexes of the model. The sparsity and running time of the experiment are taken as the other two evaluation indicators of this experiment.

The sparsity of the user-item scoring matrix can be used to measure whether the data of the matrix is sufficient. The greater the sparsity, the more elements are missing in the matrix, the sparser the matrix, and vice the denser the matrix. The calculation of matrix sparsity is as shown in Equation (23):

$$\text{sparsity} = \frac{\text{The number of 0 in the score matrix}}{\text{Number of users} \times \text{number of movies scored}}. \tag{23}$$

5 RESULT

5.1 SVD Results

Experimental results include RMSE, MAE, run time, and matrix sparsity. The results are shown in Table 3.

The results are as follows:

1. Sparsity change of the matrix: Before and after adding the user OCEAN model, the sparsity of the matrix decreased from 99.44 % to 92.64 % to 98.75 %, down from 0.69 to 6.8 percentage points.
2. Runtime change: Using the SVD-based collaborative filtering method, running time went down from 21.17600 to 0.01200 $\sim$ 4.37200 before and after joining the user OCEAN model, about 0.57 % $\sim$ 20.65 % of the original value; using the traditional collaborative filtering method, the running time decreased ranges from 11.04798 to 0.000598 $\sim$ 2.05598 before and after adding the user OCEAN model, about 0.05 % $\sim$ 18.61 % of the original value.

	All	Class Cluster					
	Users	0	1	2	3	4	5
Number of users	2012	202	391	260	599	245	315
Sparsity	99.44 %	97.44 %	98.75 %	98.11 %	98.70 %	92.64 %	97.56 %
Running time(s) (SVD)	21.176	0.012	2.703	0.110	4.372	0.013	0.613
Running time(s) (CF)	11.047	0.005	1.808	0.085	2.055	0.005	0.206
RMSE (SVD)	0.965	0.972	0.920	0.749	0.837	0.958	0.851
RMSE (CF)	0.988	0.995	0.954	0.821	0.884	0.979	0.885
MAE (SVD)	0.747	0.691	0.706	0.644	0.697	0.746	0.650
MAE (CF)	0.802	0.754	0.759	0.698	0.741	0.784	0.709

Table 3. Comparison of experimental results between improved algorithm and traditional algorithm

3. RMSE value change: Using the SVD-based collaborative filtering method, the RMSE value was 0.96526 before joining the user OCEAN model, when adding the user OCEAN model, in addition to cluster class 0 being elevated to 0.97216, the remaining clusters were decreased, the mean value was 0.881548, with an average decline of 8.67 %; Using the traditional collaborative filtering methods, the RMSE value was 0.98852 before joining the user OCEAN model, after adding the user OCEAN model, in addition to cluster 0 elevated to 0.99549, the remaining clusters were decreased, the mean value was 0.920187, with the average decline was 6.91 %.
4. MAE value change: Using the SVD-based collaborative filtering method, the MAE value decreased from 0.74766 to 0.64454 $\sim$ 0.74624, the mean value was 0.689535, with the average decline being 7.78 %; and using traditional collaborative filtering method, MAE value decreased from 0.80254 to 0.69825 $\sim$ 0.78428, the mean value was 0.741338, with the average decline was 7.63 %.

5.2 BiasSVD Results

This experiment was conducted based on the optimal parameters from the above two experimental results. The clustering was carried out based on the user OCEAN model, and the SVD collaborative filtering method and BiasSVD collaborative filtering method were used for the experiment, respectively. The experimental results included RMSE, MAE, running time, and matrix sparsity. The results are shown in the Table 4.

The results are as follows:

1. Sparsity change of the matrix: Before and after adding the user OCEAN model, the sparsity of the matrix decreased from 99.44 % to 92.64 % to 98.75 %, down from 0.69 to 6.8 percentage points.
2. Runtime change: Using the BiasSVD-based collaborative filtering method, running time went down from 10.588 to about 0.006 $\sim$ 2.186 before and after

	All	Class Cluster					
	Users	0	1	2	3	4	5
Number of users	2012	202	391	260	599	245	315
Sparsity	99.44 %	98.11 %	98.75 %	97.44 %	98.70 %	92.64 %	97.56 %
time(s) (SVD)	10.588	0.055	1.352	0.006	2.186	0.006	0.306
time(s) (BiasSVD)	0.983	0.042	0.087	0.003	0.115	0.002	0.064
RMSE (SVD)	0.965	0.962	0.920	0.759	0.837	0.958	0.851
RMSE (BiasSVD)	0.948	0.950	0.900	0.748	0.815	0.929	0.824
MAE (SVD)	0.741	0.691	0.706	0.644	0.697	0.746	0.650
MAE (BiasSVD)	0.728	0.720	0.728	0.654	0.708	0.765	0.672

Table 4. Regularization parameter selection table

joining the user OCEAN model, about 0.06 % $\sim$ 20.65 % of the original value; using the BiasSVD-based collaborative filtering method, the running time decreased ranges from 0.98399 to 0.002 $\sim$ 0.11599 before and after adding the user OCEAN model, about 0.41 % $\sim$ 11.79 % of the original value.

3. RMSE value change: Using the SVD-based collaborative filtering method, the RMSE value was 0.96528 before joining the user OCEAN model, when adding the user OCEAN model, the mean value was 0.881548, with an average decline of 8.67 %; Using the BiasSVD-based collaborative filtering methods, the RMSE value was 0. 94847 before joining the user OCEAN model, after adding the user OCEAN model, in addition to cluster 0 elevated to 0.95075, the remaining clusters were decreased, the mean value 0.861522, with the average decline, was 9.17 %.
4. MAE value change: Using the SVD-based collaborative filtering method, the MAE value decreased from 0.74766 to an average value of 0.689535, with the average decline being 7.03 %; and using the BiasSVD-based collaborative filtering method, MAE value decreased from 0.72895 to an average value 0.70839, with the average decline was 2.82 %.

6 DISCUSSION

6.1 The Comparison Between the SVD Algorithm with and without the OCEAN Model

It can be seen from the above results that the values of RMSE and MAE are correspondingly reduced, and the improved prediction effect is better after adding the user's OCEAN model in the traditional collaborative filtering method or the collaborative filtering based on SVD, compared with the result of collaborative filtering without adding the user OCEAN model. This is because the experiment considers the user's personality factor and clusters users with similar personalities into a cluster. These users have more similar preferences for movies, so the accuracy of the predictions is higher, making the recommended movies more in line with the user's preferences.

And after clustering according to the user's OCEAN model, the sparsity of the user-item scoring matrix of each cluster is lower than before clustering.

Improvements have also been made in the runtime of the recommendation system, and after a user cluster of similar personalities, it has a faster recommendation speed. As the number of system users expands, the number of users and items increases substantially, and the search and recommendation time become longer in the entire user space. However, the method proposed in this paper only needs to cluster according to the user's personality. When the target user arrives, it judges the cluster where he is located and then searches in smaller user space, the running time becomes shorter, and the efficiency of the micro-directional communication system is improved.

Although the recommendation time of traditional collaborative filtering is shorter than the recommendation time based on SVD collaborative filtering, the recommendation effect of SVD-based collaborative filtering is better from the RMSE and MAE indicators.

6.2 The Comparison Between SVD-Based and BiasSVD-Based Algorithm

Compared with the SVD-based collaborative filtering algorithm, the RMSE and MAE of the BiasSVD-based collaborative filtering algorithm are smaller, with better accuracy and much improved time efficiency. And according to the results of Table 4, the RMSE and MAE across the user space are higher than the RMSE and MAE for each class cluster, which also verifies the improvement of adding the OCEAN model in recommendation time and recommendation efficiency.

7 CONCLUSION AND PERSPECTIVE

This paper introduces a collaborative filtering method based on SVD, then proposes a method of collaborative filtering based on user OCEAN model clustering. In the experiment, the traditional collaborative filtering method, SVD-based collaborative filtering, and BiasSVD-based collaborative filtering were used to recommend users, and the proposed model was compared with the collaborative filtering model without considering the user personality model. It is verified that after adding the user's OCEAN model, a certain optimization effect can be obtained in the collaborative filtering method.

In this paper, the improved micro-directional propagation algorithm is tested. Then the parameters of the algorithm are optimized and compared with the basic algorithm before the improvement. It is proved that the algorithm of this paper has a certain improvement in the sparseness of the scoring matrix, the recommendation speed and the recommendation accuracy, which not only illustrates the importance of the user's personality model in the micro-directional communication system but also improves the performance of the micro-directional propagation system.

For the micro-directional transmission system, the model research that combines the user's OCEAN model and the micro-directional transmission system has not been widely carried out. This paper makes a preliminary exploration of the research of this model. In addition, in the following two aspects, further research can be produced in the future:

1. In the study of user OCEAN model recognition, in addition to text feature information, some non-text feature information, such as pictures, videos, etc., can be added.
2. Factors such as the user's personality model and the popularity of the item will change over time. When doing micro-targeted communication, a timestamp factor is added to the user and the item.

REFERENCES

[1] Li, X.—Liu, H.—Wang, W.—Zheng, Y.—Lv, H.—Lv, Z.: Big Data Analysis of the Internet of Things in the Digital Twins of Smart City Based on Deep Learning. Future Generation Computer Systems, Vol. 128, 2022, pp. 167–177, doi: 10.1016/j.future.2021.10.006.

[2] Khan, A. R.—Kashif, M.—Jhaveri, R. H.—Raut, R.—Saba, T.—Bahaj, S. A.: Deep Learning for Intrusion Detection and Security of Internet of Things (IoT): Current Analysis, Challenges, and Possible Solutions. Security and Communication Networks, Vol. 2022, 2022, Art. No. 4016073, doi: 10.1155/2022/4016073.

[3] Han, J.—Kamber, M.—Pei, J.: Data Mining: Concepts and Techniques. Morgan Kaufmann, 2012, doi: 10.1016/C2009-0-61819-5.

[4] Wu, Y.—Ngai, E. W. T.—Wu, P.—Wu, C.: Fake News on the Internet: A Literature Review, Synthesis and Directions for Future Research. Internet Research, Vol. 32, 2022, No. 5, pp. 1662–1699, doi: 10.1108/INTR-05-2021-0294.

[5] Prikhodko, O. V.—Cherdymova, E. I.—Lopanova, E. V.—Galchenko, N. A.—Ikonnikov, A. I.—Mechkovskaya, O. A.—Karamova, O. V.: Ways of Expressing Emotions in Social Networks: Essential Features, Problems and Features of Manifestation in Internet Communication. Online Journal of Communication and Media Technologies, Vol. 10, 2020, No. 2, Art. No. e202010, doi: 10.29333/ojcmt/7931.

[6] Cui, Z.—Xu, X.—Xue, F.—Cai, X.—Cao, Y.—Zhang, W.—Chen, J.: Personalized Recommendation System Based on Collaborative Filtering for IoT Scenarios. IEEE Transactions on Services Computing, Vol. 13, 2020, No. 4, pp. 685–695, doi: 10.1109/TSC.2020.2964552.

[7] Wang, J. L.—Jackson, L. A.—Zhang, D. J.—Su, Z. Q.: The Relationships Among the Big Five Personality Factors, Self-Esteem, Narcissism, and Sensation-Seeking to Chinese University Students' Uses of Social Networking Sites (SNSs). Computers in Human Behavior, Vol. 28, 2012, No. 6, pp. 2313–2319, doi: 10.1016/j.chb.2012.07.001.

[8] Hui, B.—Zhang, L.—Zhou, X.—Wen, X.—Nian, Y.: Personalized Recommendation System Based on Knowledge Embedding and Historical Behavior. Applied Intelligence, Vol. 52, 2022, No. 1, pp. 954–966, doi: 10.1007/s10489-021-02363-w.

[9] Sharma, S.—Rana, V.—Kumar, V.: Deep Learning Based Semantic Personalized Recommendation System. International Journal of Information Management Data Insights, Vol. 1, 2021, No. 2, Art. No. 100028, doi: 10.1016/j.jjimei.2021.100028.

[10] Singh, M.: Scalability and Sparsity Issues in Recommender Datasets: A Survey. Knowledge and Information Systems, Vol. 62, 2020, No. 1, pp. 1–43, doi: 10.1007/s10115-018-1254-2.

[11] Ma, Y.—Li, J.—Wu, X.—Yan, C.—Sun, J.—Vuduc, R.: Optimizing Sparse Tensor Times Matrix on GPUs. Journal of Parallel and Distributed Computing, Vol. 129, 2019, pp. 99–109, doi: 10.1016/j.jpdc.2018.07.018.

[12] Ding, Y.—Tian, X.—Yin, L.—Chen, X.—Liu, S.—Yang, B.—Zheng, W.: Multi-Scale Relation Network for Few-Shot Learning Based on Meta-Learning. In: Tzovaras, D., Giakoumis, D., Vincze, M., Argyros, A. (Eds.): Computer Vision Systems (ICVS 2019). Springer, Lecture Notes in Computer Science, Vol. 11754, 2019, pp. 343–352, doi: 10.1007/978-3-030-34995-0_31.

[13] Petrenko, S. A.—Stupin, D. D.: Assignment of Semantics Calculations in Invariants of Similarity. 2017 4th International Conference on Engineering and Telecommunication (EnT), IEEE, 2017, pp. 127–129, doi: 10.1109/ICEnT.2017.34.

[14] Lever, J.—Krzywinski, M.—Altman, N.: Points of Significance: Principal Component Analysis. Nature Methods, Vol. 14, 2017, No. 7, pp. 641–642, doi: 10.1038/nmeth.4346.

[15] Guo, Y.—Wang, M.—Li, X.: An Interactive Personalized Recommendation System Using the Hybrid Algorithm Model. Symmetry, Vol. 9, 2017, No. 10, Art. No. 216, doi: 10.3390/sym9100216.

[16] Saxena, A.—Prasad, M.—Gupta, A.—Bharill, N.—Patel, O. P.—Tiwari, A.—Er, M. J.—Ding, W.—Lin, C. T.: A Review of Clustering Techniques and Developments. Neurocomputing, Vol. 267, 2017, pp. 664–681, doi: 10.1016/j.neucom.2017.06.053r.

[17] Käser, T.—Klingler, S.—Schwing, A. G.—Gross, M.: Dynamic Bayesian Networks for Student Modeling. IEEE Transactions on Learning Technologies, Vol. 10, 2017, No. 4, pp. 450–462, doi: 10.1109/TLT.2017.2689017.

[18] Luo, P.—Zhang, J.—Wan, J.—Zhao, N.—Ren, Z.—Zhou, L.—Shen, J.: A Mobile Services Recommendation System Fuses Implicit and Explicit User Trust Relationships. Journal of Ambient Intelligence and Smart Environments, Vol. 13, 2021, No. 1, pp. 21–35, doi: 10.3233/AIS-200585.

[19] Marisa, F.—Ahmad, S. S. S.—Yusoh, Z. I. M.—Akhriza, T. M.—Purnomowati, W.—Pandey, R. K.: Performance Comparison of Collaborative-Filtering Approach with Implicit and Explicit Data. International Journal of Advanced Computer Science and Applications, Vol. 10, 2019, No. 10, pp. 110–116, doi: 10.14569/IJACSA.2019.0101016.

[20] Kanning, U. P.: NEO-Five-Factor-Inventory by Costa and McCrae (NEO-FFI). German Journal of Work and Organizational Psychology, Vol. 53, 2009, No. 4,

pp. 194–198, doi: 10.1026/0932-4089.53.4.194 (in German).

[21] Akritas, A. G.—Malaschonok, G. I.: Applications of Singular-Value Decomposition (SVD). Mathematics and Computers in Simulation, Vol. 67, 2004, No. 1-2, pp. 15–31, doi: 10.1016/j.matcom.2004.05.005.

[22] Bellman, R.: Introduction to Matrix Analysis. SIAM, 1997.

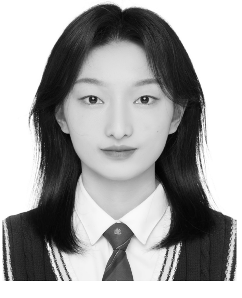
[23] Cao, J.—Hu, H.—Luo, T.—Wang, J.—Huang, M.—Wang, K.—Wu, Z.—Zhang, X.: Distributed Design and Implementation of SVD++ Algorithm for E-Commerce Personalized Recommender System. In: Zhang, X., Wu, Z., Sha, X. (Eds.): Embedded System Technology (ESTC 2015). Springer, Communications in Computer and Information Science, Vol. 572, 2015, pp. 30–44, doi: 10.1007/978-981-10-0421-6_4.

[24] Fan, J.—Ji, R.—Tian, Y.: Collaborative Filtering Recommendation Algorithm Based on Improved BiasSVD. 2021 6th International Conference on Control, Robotics and Cybernetics (CRC), IEEE, 2021, pp. 297–300, doi: 10.1109/CRC52766.2021.9620165.

[25] Cacheda, F.—Carneiro, V.—Fernández, D.—Formoso, V.: Comparison of Collaborative Filtering Algorithms: Limitations of Current Techniques and Proposals for Scalable, High-Performance Recommender Systems. ACM Transactions on the Web (TWEB), Vol. 5, 2011, No. 1, Art. No. 2, doi: 10.1145/1921591.1921593.

[26] Chai, T.—Draxler, R. R.: Root Mean Square Error (RMSE) or Mean Absolute Error (MAE). Geoscientific Model Development Discussions, Vol. 7, 2014, pp. 1525–1534, doi: 10.5194/gmdd-7-1525-2014.

[27] Yang, X.—Zhou, S.—Cao, M.: An Approach to Alleviate the Sparsity Problem of Hybrid Collaborative Filtering Based Recommendations: The Product-Attribute Perspective from User Reviews. Mobile Networks and Applications, Vol. 25, 2020, No. 2, pp. 376–390, doi: 10.1007/s11036-019-01246-2.



Yuxi Ban is a graduate student in the School of Automation at the University of Electronic Science and Technology of China, Chengdu. Her research areas include artificial intelligence, machine learning, computer vision, natural language processing, and multimodal artificial intelligence.

Yuwei Liu received her Master's degree in control science and control engineering from the University of Electronic Science and Technology of China in 2018. She is currently working as an Engineer at the Beijing Yusys Technologies Co., Ltd. Her research areas include software engineering, deep learning, natural language processing, and social network analysis.

Zhengtong Yin has been Associate Professor at the College of Resource and Environment Engineering, Guizhou University since 2009. She received her Ph.D. in Earth exploration and information technology from the Chengdu University of Technology in 2009. Her focused research interests involve artificial intelligence and machine learning. She has published more than 40 papers. Her research interests include: AI/ML, complex dynamics, pattern recognition, visual reasoning, visual question answering, NLP, surgical robot, geospatial AI, GIS/RS, image fusion, surgical vision, 3D visualization, artificial neural network, computer graphics, image processing, machine vision, 3D reconstruction, medical imaging, data mining, Earth surface process, cloud computing, geography and environmental science.

Xuan Liu is Professor at the University of Electronic Science and Technology of China. She earned her Ph.D. in urban studies in the National University of Singapore and Bachelor's degree in urban and regional studies from the Peking University. She was a visiting scholar in the Department of Architecture, Swiss Federal Institute of Technology Zürich (ETHZ), Switzerland, and in the Belk College of Business, University of North Carolina at Charlotte, USA, in 2013 and 2020, respectively. Her research focuses on patterns and mechanism of land use in urban and rural China. By applying machine learning algorithms on urban big data, she reviews and models the changes of land use patterns and space value. She also adapts various theories to explain the changes.

Mingzhe Liu received his B.Sc. degree in computer application from the Chengdu University of Technology, Sichuan, China, in 1994, and M.Sc. and Ph.D. degrees in computer science from the Massey University, New Zealand, in 2006 and 2010, respectively. He is currently Professor at the College of Computer Science and Cyber Security, Chengdu University of Technology, he is also Professor at the School of Data Science and Artificial Intelligence, Wenzhou University of Technology. His research interests include image and video processing, machine learning and deep learning, data mining, and big data. He has supervised more than 50 postgraduate students to completion. He has published over 150 peer-reviewed papers.

Lirong Yin is Ph.D. student in the Department of Geography and Anthropology at the Louisiana State University with a study interest in remote sensing, sever weather and climate change, coastal environment, natural hazard, and coupled human and natural dynamic system. She acquired her M.Sc. in geography from the Louisiana State University and her B.Sc. of Science in geography information science from the University of Iowa, she has experiences in the artificial intelligence studies and machine learning techniques, geo-data processing and information analysis skills. She has published more than 70 papers.

Xiaolu Li received her Ph.D. degree in detection technology and automatic equipment from the University of Electronic Science and Technology of China, Chengdu, China, in 2015. She is currently a Lecturer of geographic information science with the School of Geographical Sciences, Southwest University, Chongqing, China. Her research interests include the spatio-temporal information mining and coupled human and natural system.

Wenfeng Zheng is Associate Professor at the School of Automation of the University of Electronic Science and Technology of China since 2008. He received his Ph.D. from the Chengdu University of Technology in 2008. His focused research interests involve artificial intelligence and machine learning. He has published more than 160 papers, and he authorized more than 50 Chinese national invention patents. He is a member of the Association for Computing Machinery, IEEE, American Association of Geographers, American Geophysical Union, and China Association of Inventions. He is ranked among the 2022 World's top 2 % scientists list of the Stanford University.