Hindawi
Journal of Mathematics
Volume 2023, Article ID 9861405, 1 page
https://doi.org/10.1155/2023/9861405

Retraction

Retracted: Intelligent Recommendation Algorithm of Multimedia English Distance Education Resources Based on User Model

Journal of Mathematics

Received 19 December 2023; Accepted 19 December 2023; Published 20 December 2023

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope

(2) Discrepancies in the description of the research reported

(3) Discrepancies between the availability of data and the research described

(4) Inappropriate citations

(5) Incoherent, meaningless and/or irrelevant content included in the article

(6) Manipulated or compromised peer review

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] X. Zhang, "Intelligent Recommendation Algorithm of Multimedia English Distance Education Resources Based on User Model," *Journal of Mathematics*, vol. 2022, Article ID 2012700, 8 pages, 2022.

Hindawi
Journal of Mathematics
Volume 2022, Article ID 2012700, 8 pages
https://doi.org/10.1155/2022/2012700

Research Article

Intelligent Recommendation Algorithm of Multimedia English Distance Education Resources Based on User Model

Xiushan Zhang

Chongqing City Vocational College, Yongchuan, Chongqing 402160, China

Correspondence should be addressed to Xiushan Zhang; 20040022@cqwu.edu.cn

Received 23 November 2021; Revised 19 December 2021; Accepted 21 December 2021; Published 13 January 2022

Academic Editor: Naeem Jan

Based on the understanding and comparison of various main recommendation algorithms, this paper focuses on the collaborative filtering algorithm and proposes a collaborative filtering recommendation algorithm with improved user model. Firstly, the algorithm considers the score difference caused by different user scoring habits when expressing preferences and adopts the decoupling normalization method to normalize the user scoring data; secondly, considering the forgetting shift of user interest with time, the forgetting function is used to simulate the forgetting law of score, and the weight of time forgetting is introduced into user score to improve the accuracy of recommendation; finally, the similarity calculation is improved when calculating the nearest neighbor set. Based on the Pearson similarity calculation, the effective weight factor is introduced to obtain a more accurate and reliable nearest neighbor set. The algorithm establishes an offline user model, which makes the algorithm have better recommendation efficiency. Two groups of experiments were designed based on the mean absolute error (MAE). One group of experiments tested the parameters in the algorithm, and the other group of experiments compared the proposed algorithm with other algorithms. The experimental results show that the proposed method has better performance in recommendation accuracy and recommendation efficiency.

1. Preface

With the promotion of the national policy of coconstruction and sharing of educational information resources, the number and types of educational resources are unprecedented rich, and the improvement of people's cognitive ability lags far behind the speed of information diffusion. However, massive educational information resources cause cognitive overload, information trek, and anxiety problems. Learners' access to personalized learning resources is like looking for a needle in a haystack, and learning everywhere has evolved into search everywhere [1–3]. How to reduce the information search cost and annoyance cost of learners, so that learners with different information literacy can obtain information resources suitable for their own needs, and provide educational resource services in line with their personality development needs for learners with different knowledge structures and intelligence types has become an unavoidable practical problem [4, 5].

According to the different types of learning resources, learning objectives, and student groups, designing a flexible personalized recommendation model of learning resources has become a breakthrough to solve this educational problem [6, 7]. This study proposes an intelligent recommendation of educational resources based on user model, which aims to mine educational resources and learning partners that meet the individual needs of learners from massive educational data, recommend learning activities that adapt to learners' cognitive styles, and provide them with adaptive and personalized educational services [8].

2. Algorithm Proposed

Accurate representation of educational resources and effective knowledge organization are necessary prerequisites for the intelligent recommendation of multimedia English distance education resources [9]. To truly realize the coconstruction, sharing, and extensive benefits of multimedia English distance education resources, on the one hand, it is necessary to accurately and comprehensively describe educational resources; on the other hand, it is necessary to effectively screen and organize educational resources. Collaborative filtering recommendation technology can solve the above problems. It is to find neighbors with similar interests to the target user and predict the preferences of the target user from the interest preferences of the neighbor users, thereby completing the course recommendation for the target user [10].

The basic idea of collaborative filtering recommendation is based on these two assumptions. One is that if two users have very similar hobbies, then the course that one user likes is likely to be liked by the other user; the other is if two courses are very similar, then the user likes one course at the same time; it is very likely that they will also like the other course. Collaborative filtering can be divided into memory-based filtering and model-based filtering. Memory-based collaborative filtering calculates recommendations on all data sets, and each calculation adds new data to the calculation again. With the continuous increase of data, the scalability of this memory-based collaborative filtering recommendation system is greatly reduced. At present, some new technologies, such as clustering, Bayesian networks, machine learning, and association rules, have also been applied to the establishment of the model, in order to better improve the quality of collaborative filtering recommendations. In short, collaborative filtering recommendation is currently attracting the attention and favor of recommendation method researchers [11].

In response to the above problems, Deng [12] used Gaussian distribution method to normalize the score. Yitao et al. [13] proposed a decoupling normalization method and concluded that this method has better performance than Gaussian normalization. This paper uses this decoupling normalization method to process user ratings. The collaborative filtering algorithm proposed in this paper is based on an improved user model. It is called an improved user model because the user score is normalized before modeling with a user score matrix. Many models of collaborative filtering recommendation are established based on user ratings, but user ratings will vary due to the habits of different users. Users who have common interests in the same course may give different ratings to the course. At the same time, referring to the famous Ebbinghaus forgetting curve, a nonlinear logistic function is designed for the algorithm to explore the user's rating forgetting rule more closely, so as to give each normalized rating a different time forgetting weight [14–16]. Considering the influence of the recommendation and the authenticity of the neighbor set, this paper sets an effective weight factor when calculating the user similarity, which helps to improve the accuracy of the recommendation result [17].

3. Improved Collaborative Filtering Algorithm of User Model

The traditional collaborative filtering recommendation calculation of user similarity is based on the original user score data, and the recommendation quality is not very ideal [18]. Based on the basic idea of traditional collaborative filtering recommendation, the collaborative filtering algorithm of improved user model in this paper improves the recommendation efficiency by offline modeling and online recommendation. At the same time, the normalization of user score and the introduction of scoring time weight in the establishment of the model fully consider the different scoring habits of users and the fact that user interests shift with time, so as to ensure the quality of online recommendation in the next step. During online recommendation, the built user model is loaded into memory, and the nearest neighbor set of the target user is generated by calculating the user similarity. These nearest neighbor sets will be used as the reference users who have the most similar interest preferences with the target user, participate in predicting the score of the target user on the nonscored items, and sort the predicted scores. In this way, the collaborative filtering recommendation result based on the improved user model is generated. When calculating the user similarity, this paper fully considers the impact of the number of user common scoring items on the user similarity and excludes the deviation of the recommendation results caused by the fact that there are few items scored by two users and the calculated user similarity is very high. The workflow of the collaborative filtering recommendation algorithm proposed in this paper is shown in Figure 1.

In order to improve the traditional user model, the algorithm in this paper first uses the decoupling method to normalize the user score and then uses the forgetting function to assign different time forgetting weights to the score according to the theory of Ebbinghaus forgetting curve and then processing in this way. The user similarity is calculated on the latter scoring matrix for recommendation.

3.1. Normalized Score.

Many collaborative filtering recommendations are based on user rating data, so user rating represents users' real interests and hobbies to ensure the accuracy of recommendations. In the further study of user ratings, this paper finds that the original user rating data show that users with the same interest preferences have differences in ratings:

(i) Different scoring ranges: some users prefer to score in a larger range, while others prefer to score in a smaller range

(ii) Different scoring scales: some users are more "tolerant," it is easier to "show mercy" when scoring, and the score is generally high; on the contrary, some users do not give the highest score even if they like it

Because users have such habit differences in scoring, this paper reduces its impact on the recommendation effect and uses the decoupling method to normalize users' scores. Decoupling normalization is a method of probability mechanism, which is based on two assumptions:

One is that if most of the user's course scores are less than or equal to R, it means that the user is likely to like the course;

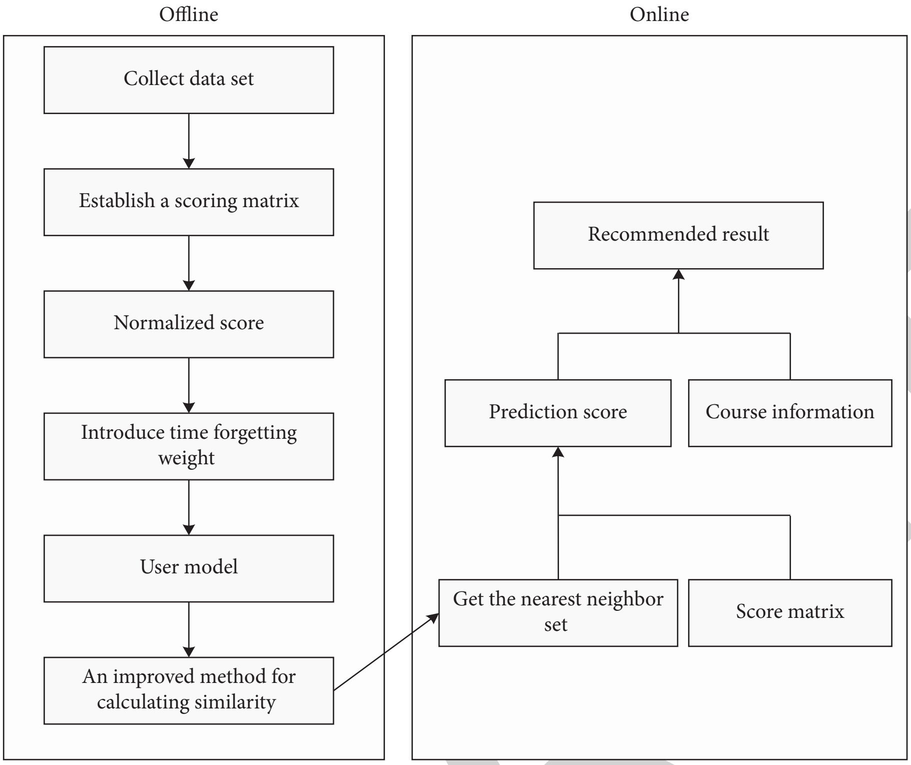
Offline
Online
Collect data set
Establish a scoring matrix
Normalized score
Introduce time forgetting weight
User model
An improved method for calculating similarity
Recommended result
Prediction score
Course information
Get the nearest neighbor set
Score matrix

Figure 1: The workflow of collaborative filtering recommendation algorithm with improved user model.

The second is that if user i scores a large part of courses with score R, then the course with score R is less likely to be liked by user i.

Based on these two assumptions, according to the semiaccumulative distribution method, the equation for defining decoupling normalization is as follows:

$$\widetilde{R}_{i,j} = P_i(R \,\text{is preferred}) = \frac{P_i(\text{Rating} \le R) - P_i(\text{Rating} = R)}{2}. \tag{1}$$

In the equation, $P_i(R \,\text{is preferred})$ is the result of this normalized processing scoring, which represents the probability that the course scored R is liked by the user i. R represents a rating level, $P_i(\text{Rating} \le R)$ and $P_i(\text{Rating} = R)$ respectively represent the probability that the user i will score a course less than or equal to R and equal to R, reflecting the probability that a course rated less than or equal to R and a course equal to R will be liked by the user i. In this way, for the course j scored by the user i as R, we can use $P_i(R \,\text{is preferred})$ as the normalized result of the user i which is the rating of the course j and mark it as $\widetilde{R}_{i,j}$.

3.2. Introducing Time Forgetting Weight. With the change of time, users' interests will always shift. Therefore, the scores given by users in different time periods have different reference significance for recommendations. All scores cannot be treated the same without considering the impact of time on the reference value of scores.

To illustrate this problem, let us give an example. Table 1 shows the scoring records of five courses made by four users. It should be noted that the time periods of these scores are somewhat different.

In Table 1, if the time when the score is generated is not considered, then from the score point of view, it is easy to get $SIM(U_1, U_2) > SIM(U_1, U_3) > SIM(U_1, U_4)$, that is, the nearest neighbor of user U_1 is U_2. However, when we consider the time period in which the score was generated, the result has changed. At this time, you will find that it is unreasonable to use U_2 as the nearest neighbor of U_1, because the score of a user in the past time period is used to calculate the similarity with the score of another user in the current time period to explore whether they are similar. Interest preferences are meaningless. Therefore, considering the impact of the scoring time period on the scoring, the nearest neighbor of U_1 in Table 1 should be U_4.

This is an example of considering the impact of time forgetting on scoring between different users, and for a certain user, the user interest shift caused by time forgetting also exists. The user's past preferences will change over time. Past ratings and current ratings cannot be treated the same, and different time weights should be assigned. Next, an appropriate function must be selected to simulate the user's interest shift law to generate time forgetting weights.

In fact, the deviation rule of user interest is very similar to the forgetting rule of people, so the time forgetting function can be given by referring to the forgetting rule. Regarding the study of the law of forgetting, the results of the German psychologist Ebbinghaus are worth learning. The famous Ebbinghaus forgetting curve has been cited in many researches. The Ebbinghaus forgetting curve shows the nonlinear decreasing law of human memory retention. The

Table 1: Scores in different scoring time periods.

	I1	I2	I3	I4	I5	Time period
U_1	2	3	3	4	5	T3
U_2	2	2	3	—	5	T1
U_3	—	3	3	3	4	T2
U_4	1	—	2	4	4	T3

algorithm in this paper uses a logistic function to simulate this curve law [19], reflecting the forgetting shift trend of user interest. The logistic function model is shown in Figure 2.

This logistic function is nonlinearly increasing, which means that the longer the time passes, the more the interest forgetting shifts. However, considering that the deviation of user interests and hobbies is a bit different from forgetting, user interests and hobbies remain relatively stable during a period of time and will not change quickly. The algorithm has made appropriate improvements to the logistic function. Here is a definition: the parameters of the time gap make users' ratings in the same time gap produce the same forgetting function value, which is more in line with the interest shift law. Therefore, the time forgetting function is defined as

$$
\begin{aligned}
f\left(t_{i,j}\right) &= \frac{1}{1+e^{-\left(\left(tl - t_{i,j}\right)/T\right)}},\\
f\left(t_{i,j}\right) &= \frac{1}{(1/K)+b*e^{tl}},\\
b &= \frac{e^{a}}{K},\\
b &= \frac{e^{a}}{K},
\end{aligned}
\tag{2}
$$

In the function, $t_{i,j}$ represents the time when user i scored course j, which is the independent variable of the function, and tl represents the time when the user scored the last time. T defines a time gap parameter, and its value depends on the result of experimental verification. a, b are constants greater than zero. K is the theoretical limit value. Use the fitting method to solve the parameters; the formula is

$$
\phi = \sum_{i=1}^{n}\left(t_{i,j} - \widehat{t}_{i,j}\right)^2 = \sum_{i=1}^{n}\left(t_{i,j} - (a - bT)\right)^2. \tag{3}
$$

In the function, n is the number of samples. Calculate the partial derivative of the constant a, b according to the ϕ function and make the partial derivative equal to 0:

$$
\begin{aligned}
\frac{\partial\phi}{\partial a} &= 0,\\
\frac{\partial\phi}{\partial b} &= 0.
\end{aligned}
\tag{4}
$$

Therefore, the unknown parameter a, b is obtained, and the time forgetting weight can be defined as

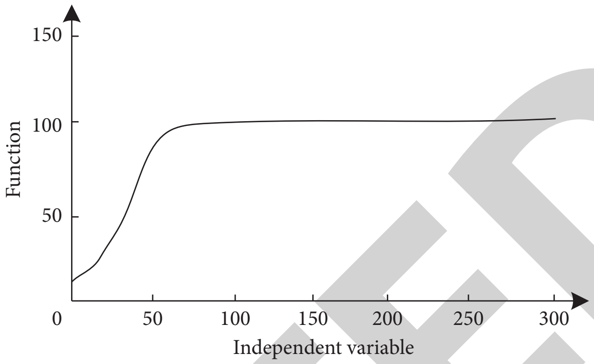

Figure 2: Logistic function model curve.

$$
W_{t_{i,j}} = w\left(1 - f\left(t_{i,j}\right)\right). \tag{5}
$$

Among them, w is the specification parameter. The time forgetting weight is introduced into the score result of the normalization process, and the processed score is recorded as $R_{i,j}$, which is expressed as

$$
R_{i,j} = \widetilde{R}_{i,j} \times W_{t_{i,j}.} \tag{6}
$$

So far, the user scoring matrix obtained after processing is the user model established by the algorithm in this paper.

In general, the collaborative filtering recommendation based on the model comprises of the following steps:

Step 1: data collection

Step 2: building the model

Step 3: finding the nearest neighbor collection

Step 4: recommending the forecast

The previous content has discussed the establishment process of the model in the collaborative filtering algorithm to improve the user model. The next work is mainly to recommend based on the model, that is, to find the nearest neighbor set and predict the recommendation.

3.3. Find the Nearest Neighbor Set. The process of finding the set of nearest neighbors of the target user is actually the process of calculating the user similarity between the target user and other users. The two users are similar, which here means that the two users have similar interests and preferences. In the collaborative filtering recommendation based on the user model, the similarity between users is relatively large, and they are more likely to like the same course. Therefore, the user similarity can be used to find the set of nearest neighbors of the target user. Like courses, you can predict which courses the target users are more interested in, so as to make personalized recommendations for the courses.

There are several similarity calculation methods, and there are three widely used in collaborative filtering recommendation algorithm, namely, Pearson correlation coefficient method [20], cosine similarity, and improved cosine similarity method. The algorithm in this paper uses Pearson's correlation coefficient method to calculate user similarity. The similarity calculation formula of Pearson's correlation coefficient is defined as follows:

$$\mathrm{sim}\,(a,b)=\frac{\sum_{j\in I_{ab}}\left(R_{a,j}-\overline{R_a}\right)\left(R_{b,j}-\overline{R_b}\right)}{\sqrt{\sum_{j\in I_{ab}}\left(R_{a,j}-\overline{R_a}\right)^2}\sqrt{\sum_{j\in I_{ab}}\left(R_{b,j}-\overline{R_b}\right)^2}}, \tag{7}$$

where $\mathrm{sim}\,(a,b)$ represents the similarity between user a and user b, and $R_{a,j}$ represents user a which is the rating of course j. In this algorithm, this rating is the rating processed in the processing model. Similarly, $R_{b,j}$ represents user b which is the rating of course j. $\bar{R_a}$ represents the average of all ratings of user a, and $\bar{R_b}$ represents the average of all ratings of user b. $I_{a,b}$ refers to the collection of courses that the user a and the user b have jointly rated.

The purpose of calculating the user similarity is to find the real nearest neighbor of the target user. However, when calculating the user similarity according to formula 5, it may occur that the two users only have few common scoring courses, but when calculating the similarity, it happens that the similarity between the two users is large, which overestimates the similarity between the two users. Using such nearest neighbors to predict the courses that target users are interested in, the quality of recommendation after prediction will be affected.

In order to improve the accuracy of similarity calculation and improve the above problems, an effective weight factor is proposed to improve Pearson correlation similarity calculation. This effective weight factor is defined as follows:

$$w=\frac{\mathrm{Min}\,(\,|I_{a,b}|,\delta)}{\delta}, \tag{8}$$

where $|I_{a,b}|$ represents the number of courses scored by user a and user b, and δ is an adjustable parameter. Use it to set the threshold for the number of courses scored by user a and user b. In the experiment, we will find one for parameter δ appropriate value. This effective weight factor is to give the similarity value calculated by the Pearson similarity degree. It can be explained in this way. If the number of courses that two users have jointly rated exceeds the set threshold δ, the weight is 1. The user similarity of two users depends on the result of Pearson's correlation similarity calculation; on the contrary, if the number of courses scored by two users does not exceed the threshold, this effective weight will play its role. Obviously, the two users share the same. The fewer the number of courses that have been rated, the smaller the numerator, and the smaller the effective weight factor. That is to say, at this time, the contribution of the Pearson correlation similarity calculation result to the similarity value between the end users will decrease.

By adding this effective weight factor, the final user similarity calculation formula can be expressed as

$$\mathrm{sim}'\,(a,b)=w\times\mathrm{sim}\,(a,b). \tag{9}$$

In the above equation, $\mathrm{sim}\,(a,b)$ represents the similarity between users a and b, w indicates the weight factor, and $\mathrm{sim}'\,(a,b)$ represents the final similarity after the addition of weight factor's effectiveness. After calculating the similarity between the target user and other users, the score of the target user on historical items is introduced to further revise the similarity formula:

$$\mathrm{sim}_0'(a,b)=\overline{r}+\mathrm{sim}'\,(a,b), \tag{10}$$

where $\overline{r}$ is the average score of the item that the user has rated. At this time, the top M users with the highest similarity value can be selected as the nearest neighbor set of the target user, and the nearest neighbor set helps the target user predict and recommend courses that may be liked. The size M of the nearest neighbor set here is determined by the specific recommendation background. In the experiment, the optimal value of the collaborative filtering recommendation algorithm of the improved user model in the recommendation background and experimental data environment of this article is also obtained [21].

3.4. Forecast Recommendation. After obtaining the set of nearest neighbors of the target user, this step is to predict the score of the target user for the scoring course based on the set of nearest neighbors. This step uses the traditional collaborative filtering recommendation algorithm. The prediction score is partly determined by the average score of the target user and partly determined by the neighbors of the nearest neighbor set. The formula for predicting user a which is the rating of ungraded course and j is as follows:

$$p\,(a,j)=\overline{R_a}+\frac{\sum_{i\in UN_a}\mathrm{sim}\,(a,i)\times\left(R_{i,j}-\overline{R_i}\right)}{\sum_{i\in UN_a}\mathrm{sim}\,(a,i)}, \tag{11}$$

where $\bar{R_a}$ represents the average score of user a and UN_a represents the set of nearest neighbors of user a.

By calculating the target user's score for the scoring course, the obtained top N quotients with higher predicted scores can be recommended to the user as the final personalized recommendation result of the target user.

4. Data Set and Experimental Measurement

4.1. Experimental Data Set. Most of the experimental data sets of collaborative filtering recommendation algorithms are derived from some well-known university multimedia English distance education resource recommendation systems.

The research of many algorithms of collaborative filtering recommendation is based on the data set of the recommendation system. According to the research environment and conditions of the laboratory, this paper uses the ml data set of Northeastern University as the experimental data set. This data set contains 100000 rating records recommended by 943 users for 1682 English-related content, with a rating range of 1–5, and each user has at least 20 rating records. In the experiment, 80% of the data is used as training data, and the remaining 20% is used as experimental verification data.

4.2. Experimental Environment and Tools. The environment of this experiment is Microsoft Windows 7 + Java Develop Kit v1.6.0 + SQL Server 2008, the simulation system runs on the Tomcat 6.0 platform, and the server configuration is Intel(R) Xeon(TM) CPU 2.80 GHz, 2 GB memory.

4.3. Experimental Measurement

4.3.1. Experimental Measurement Standards. There are several standards to measure the accuracy of collaborative filtering algorithm. In this paper, the average absolute error (MAE) is used to measure and verify the accuracy of the algorithm.

The mean absolute error (MAE) calculates the mean absolute error between the predicted score and the real score. The accuracy of the algorithm is judged by the size of the error difference.

If I_a is used to represent the set of courses with both predicted and true ratings for the target user, $p_{a,i}$ is the predicted score of the target user *a* for the course *i*, and $r_{a,i}$ is the true score of the target user *a* for the course *i*. The calculation method of error (MAE) can be defined as

$$MAE = \frac{\sum_{i \in I_a} \left| p_{a,i} - r_{a,i} \right|}{\left| I_a \right|}. \tag{12}$$

The smaller the calculated MAE value, the higher the accuracy of the algorithm.

The following will take the mean absolute error (MAE) as the measurement standard, divided into two groups of experiments to measure and verify the algorithm proposed in this paper.

4.3.2. Parameter Setting. In the algorithm of this paper, there are two adjustable parameters that need to be set through experiments. One is the time gap parameter T in the logistic function for calculating the time forgetting weight, and the other is the δ threshold in the effective weight factor in the similarity calculation. The experimental results of the influence of these two parameters on the algorithm are shown in Figures 3 and 4.

In the experiment of the test value of the parameter T, the unit of T is taken as time weeks, and the scoring time in the scoring record is seconds. Considering that the user's interest preferences will not change in seconds, the unit of T is converted in the experiment. For a certain user, the experiment changes the value of T to observe the influence of the change of T on the average absolute error (MAE). As shown in the result of Figure 3, under the ml data set of Movie Lens, the best value of T in the algorithm of this paper is 2 weeks. Of course, the value of T will be different in different recommended environments for the algorithm.

For the value of δ in the effective weight factor, the experiment is set to change the value of δ under the condition that the value of T remains the same for a certain user and the value of T is the best value for 2 weeks, and at the same time, the value of δ is obtained for the average absolute error (MAE). The result of the trial value experiment is shown in Figure 4. This result shows that in the algorithm experiment environment of this paper, the best value range of δ is 16 to 20, and the best value of 18 is selected in the comparison experiment. Similarly, when the algorithm is used in different recommendation systems, the value of δ should also be reset.

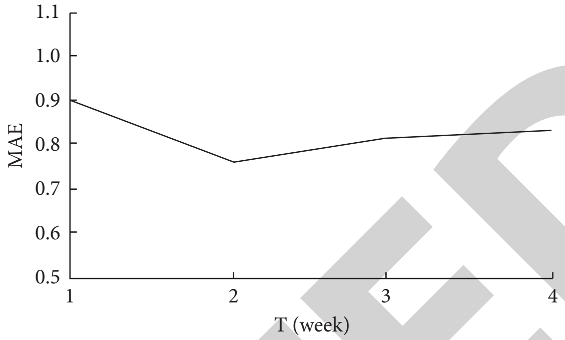

Figure 3: Parameter T test value result.

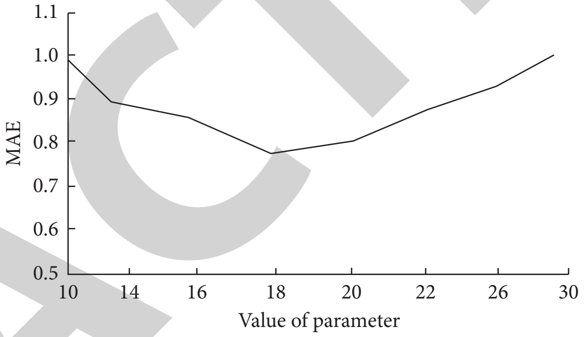

Figure 4: Parameter δ test results.

4.3.3. Comparative Measurement. Next, the proposed collaborative filtering algorithm is compared with the other two algorithms, i.e., the traditional collaborative filtering algorithm, and the collaborative filtering algorithm based on hybrid user model. These algorithms are abbreviated by English initials as follows:

> CCF: conventional collaborative filtering is a traditional collaborative filtering algorithm
>
> HUMCF: hybrid user model based collaborative filtering is a collaborative filtering algorithm based on the hybrid user model
>
> IUMCF: improved user model based collaborative filtering is the improvement of the collaborative filtering algorithm of user models

CCF is a classic of collaborative filtering algorithms, so the algorithm in this article is first compared with it; HUMCF is a collaborative filtering method based on a hybrid user model, which is based on such a hybrid user model, which combines user ratings, course features, and demographic information. Weights of feature vectors are learnt using genetic algorithms, so as to calculate the similarity between users to generate a set of nearest neighbors. Experiments have proved that this algorithm also achieves high recommendation accuracy. The author has participated in the research of this algorithm. Therefore, the experiment compared the system filtering algorithm proposed in this paper with the improved user model based on this algorithm.

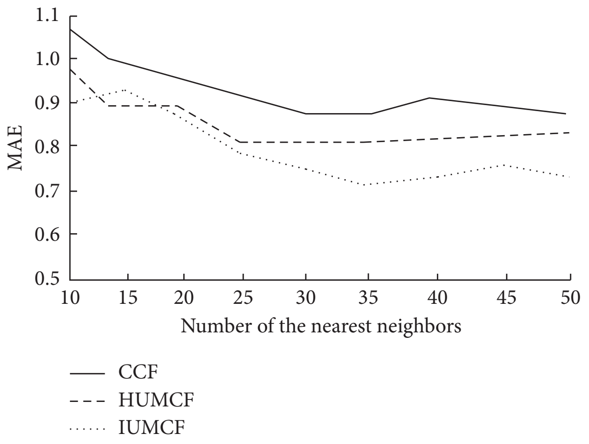

Figure 5: Comparison of MAE results of the same user with different numbers of neighbors.

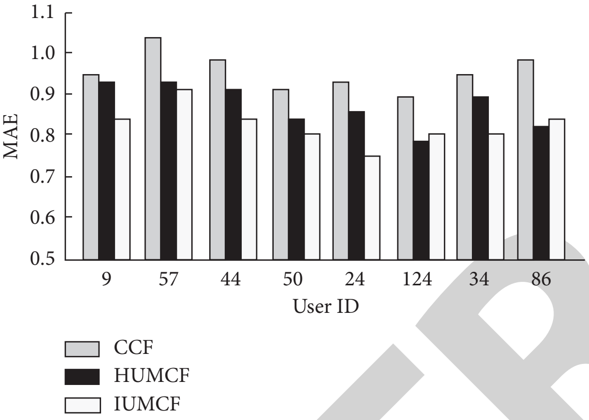

Figure 6: Comparison of MAE results under different users with the same number of neighbors.

IUMCF represents the collaborative filtering algorithm proposed in this paper to improve the user model. The experiment compares the average absolute error (MAE) results of the three algorithms under two different conditions.

First, take the best values for each parameter, and look at the changes in the MAE values of the three algorithms under different numbers of nearest neighbors. After the experiment, the results are shown in Figure 5.

Through experiments, the number of nearest neighbors ranges from 10 to 50. From the results, the collaborative filtering algorithm based on the hybrid model and the collaborative filtering algorithm based on the improved user model in this paper have lower average absolute error (MAE) than the traditional collaborative filtering algorithm. But the algorithm in this paper obviously has a lower MAE value. It can be said that the collaborative filtering algorithm proposed in this paper to improve the user model has better recommendation accuracy.

Secondly, the experiment compares the average absolute error of the three algorithms for different random users. Similarly, all parameters take the best values. The parameter T takes 2 weeks, δ takes 18, and the number of nearest neighbors takes 35. It just selects different users randomly, and different user IDs of different users are randomly selected. The experimental results are shown in Figure 6.

It can be seen from the results in Figure 6 that the MAE values of the algorithm are different for different users, but more importantly, the MAE values of the three algorithms are obviously different for different users. In general, the collaborative filtering algorithm based on hybrid model and the collaborative filtering algorithm of improved user model in this paper can still get lower average absolute error (MAE) than the traditional collaborative filtering algorithm, but the algorithm in this paper has lower MAE value, which further shows that the collaborative filtering algorithm of improved user model proposed in this paper has better performance in recommendation accuracy.

5. Conclusion

The collaborative filtering algorithm of improved user model improves the accuracy of the algorithm at the level of user ratings through the normalization of ratings and the introduction of time forgetting weights. At the same time, the effective weight factor is added when calculating user similarity. On the one hand, it can avoid the impact of data sparsity on the recommendation, and on the other hand, it can make the set of nearest neighbors to be more reasonable, thereby helping to improve the accuracy of the recommendation. The offline modeling of the algorithm and the online recommendation mode save online waiting time for recommendation and improve recommendation efficiency to a certain extent. Experiments have also proved that this improved user model collaborative filtering algorithm has a good performance in recommendation accuracy and recommendation efficiency.

Data Availability

The data used to support the findings of this study are available from the author upon request.

Conflicts of Interest

The author declares that he has no conflicts of interest.

References

[1] L. Li, "Optimal construction of the multimedia distance education resource library based on chinese education," in *Proceedings of the International Conference on Application of Intelligent Systems in Multi-Modal Information Analytics*, Springer, Cham, Switzerland, 2019.

[2] L. F. Lu, "An analysis of influential factors of motivation learning behavior of English learners in distance education," *Journal of YunNan Open University*, vol. 22, no. 1, p. 9, 2020.

[3] P. Branscum and N. Patricio-Agosto, "How children search for health information online: an observational study," *Journal of Nutrition Education and Behavior*, vol. 52, no. 5, pp. 522–527, 2020.

[4] Y. S. Xi, "Development of intelligent english multimedia teaching resources using data mining," *Heimat*, pp. 150–155, Springer, Cham, Switzerland, 2019.

[5] C. D'Souza, M. Brahme, and M. S. Babu, "Environment education in Indian schools: the search for a new language," *Journal of Education for Sustainable Development*, vol. 14, no. 2, pp. 174–189, 2020.

[6] C. Li, Y. Li, C. Wang et al., "The multimedia recommendation algorithm based on probability graphical model," *Multimedia Tools and Applications*, vol. 5, pp. 1–16, 2020.

[7] W. B. Cutrer, W. A. Spickard, M. M. Triola et al., "Exploiting the power of information in medical education," *Medical Teacher*, vol. 43, no. 2, pp. 17–24, 2021.

[8] Y. Cui, "Intelligent recommendation system based on mathematical modeling in personalized data mining," *Mathematical Problems in Engineering*, vol. 2021, no. 3, pp. 1–11, 2021.

[9] L. G. Ferreira, J. L. V. Barbosa, J. C. Gluz, V. K. Matter, and D. N. F. Barbosa, "Using learner group profiles for content recommendation in ubiquitous environments," *International Journal of Information and Communication Technology Education*, vol. 16, no. 4, pp. 1–19, 2020.

[10] Y. Chen, "Research on Personalized Recommendation Algorithm Based on User Preference in Mobile E-Commerce," *Information Systems and e-Business Management*, vol. 18, no. 3, 2019.

[11] Q. Yang, "Network-based college english multimedia teaching model," *Application of Intelligent Systems in Multi-modal Information Analytics*, Springer, Cham, Switzerland, 2021.

[12] C. Y. Deng, "Problems and countermeasures of multimedia assisted english teaching in distance education," *Journal of Hubei Correspondence University*, vol. 6, 2017.

[13] Y. Wu, X. Zhang, H. Yu, and S. Wei, "Collaborative filtering recommendation algorithm based on user fuzzy similarity," *Intelligent Data Analysis*, vol. 21, no. 2, 2017.

[14] C. Ming and T. Dan, "The multimedia technology application in the network distance education—accountancy as a model," in *Proceedings of the 2015 7th International Conference on Information Technology in Medicine and Education (ITME)*, IEEE, Huangshan, China, November 2015.

[15] E. A. M. Leite, A. Machado, G. L. Sales, H. Zednik, and S. M. Maia, "Distance education evaluation: an analysis of the β factor from LV model subjectivity," *New Contributions in Information Systems and Technologies*, vol. 353, pp. 1231–1241, 2015.

[16] I. Bartolini, V. Moscato, R. G. Pensa et al., "Recommending multimedia visiting paths in cultural heritage applications," *Multimedia Tools and Applications*, vol. 75, no. 7, pp. 3813–3842, 2016.

[17] M. Klein, M. Leitzgen, and M. Weyrich, "Description of an intelligent resource unit for a smart production," in *Proceedings of the 2016 IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA)*, IEEE, Berlin, Germany, September 2016.

[18] Y. Wu, Y. Zhao, and S. Wei, "Collaborative filtering recommendation algorithm based on interval-valued fuzzy numbers," *Applied Intelligence*, vol. 50, no. 3, pp. 2663–2675, 2020.

[19] F. B. Demir, T. Tuncer, and A. F. Kocamaz, "A chaotic optimization method based on logistic-sine map for numerical function optimization," *Neural Computing & Applications*, vol. 32, no. 17, pp. 14227–14239, 2020.

[20] W. Zeng, F. Wang, L. Miao, F. You, and F. Yao, "Laser ultrasonic melanoma detection in human skin tissues via Pearson correlation coefficient," *Optik*, vol. 222, no. 6, Article ID 165478, 2020.

[21] P. B. Adhikari, *The 8th International Conference on Computer Engineering and Networks*, Springer, Berlin, Gremany, 2018.