IUCFAMR: AN IMPROVED MOVIE RECOMMENDATION ALGORITHM

Shigan YU[1], Fujun REN[2,*]

With the advent of the big data era of information overload, recommendation technology has been integrated into modern daily life. In view of some deficiencies of data sparsity and real-time scalability in traditional recommendation algorithms, it is found through investigation that popular movies are often recommended to users frequently in some movie recommendation systems, which affects the selection of non-popular movies by some users and fails to take into account that users' interests and hobbies change with time. Therefore, this paper proposes the Improved User-based Collaborative Filtering Algorithm for Movie Recommendation (IUCFAMR). Based on the time context, the IUCFAMR proposes to add the penalty term and the time factor of context to solve these two problems. The IUCFAMR solves the problems of data sparsity and real-time scalability to a certain extent by improving the mathematical methods in the recommendation algorithm. Experiments have proved that the IUCFAMR improves the recommendation effect and accuracy for users.

Keywords: Big data; Collaborative Filtering Algorithm (CFA); Penalty term;Time factor; User similarity

1. Introduction

Nowadays, the phenomenon of information overload is becoming increasingly serious. A variety of recommendation algorithms are playing a positive role in all walks of life. For example, in daily life, if people often go to the cinema to watch movies, most of the movies in the cinema are shown in advance and the number of movies is small, so it is relatively easy for people to choose. However, due to the large number of movies in the dazzling home theater, it is relatively difficult to choose a suitable movie. In the face of the huge number of movies on the network, how to choose the movie that suits their own taste has become a difficult problem. The movie recommendation system can improve the quality of life of users.

Recommendation algorithm has attracted the attention of many researchers. For example, in order to better recommend potential candidates. Gegen Tana proposed a technology that combines partitioning clustering technology with CFA and dynamic decision graph to achieve a good recommendation effect [1]. Qiaochu Yu proposed a CFA based on the idea of optimal combination prediction [2]. Ramil

[1] College of Information Engineering, Fuyang Normal University, Fuyang 236041, Anhui, China

[2] Haiyan College, Jiaxing Vocational and Technical College, Jiaxing 314000, Zhejiang, China; Corresponding author email: rfj17792121962@163.com

G. Lumauag proposed an improved algorithm based on data set enhancement algorithm and evaluated its accuracy and performance to optimize the impact of data sparsity and overfitting on the accuracy of recommendation systems [3]. İclal ÖZCAN and Mete ÇELİK put forward an efficient recommendation system in DataProc in combination with alternative least square method and CFA to improve its recommendations in cloud computing [4]. In consideration of the influence of personal preferences on users' preferences over time, Zhang Pengfei proposed a wechat ordering system incorporating time weights [5]. User similarity is used to measure whether items are similar to each other and users to users. User similarity is that two users who have similar interactions on the same project should be considered close. In view of the influence of popular songs on users' choices, Qian Beibei proposed a CFA which added a penalty factor into the user similarity [6]. Lee Cheong Rok et al. used sigmoid function to reflect preferences and improve the calculation of similarity in system [7]. To solve the problems of sparse data and cold startup in traditional CFA, Du proposed to introduce user rating weight into the calculation of user similarity to improve the performance [8]. Jain Gourav proposes a time decay that gives more weight to recent ratings to reduce the impact of data sparsity [9]. Ulian Douglas Zanatta proposes a hybrid CFA that combines factors such as behavior, interest, article popularity and time effect, improving recommendation performance [10].

These recommendation algorithms have improved the recommendation effect to a certain extent; However, these methods still have some problems in data sparsity, cold startup and expansibility. Therefore, this paper puts forward the IUCFAMR method, adding the mathematical method of penalty term and time factor in order to better calculate the user similarity and improve the movie recommendation effect.

2. An Introduction to CFA

CFA relies on the behavioral relationship between users and items to generate recommendations, and collaborates with users' opinions, feedback, evaluation and other factors to screen and filter a large amount of information. By analyzing the similarities between users or things, it predicts the content that users may be interested in and recommends the content to users. CFA are mainly divided into three types: User-Based CFA, item-based CFA and model-based CFA [11].

2.1. User-Based CFA

The core of **CFA** is to simulate the data information from different users to the project into a vector and calculate the similar value between the two vectors. According to the similarity, find the users who are similar to the target users, and recommend the items bought by the similar users and the items not bought by the target users to the target users. The user-based **CFA** is shown in Fig.1.

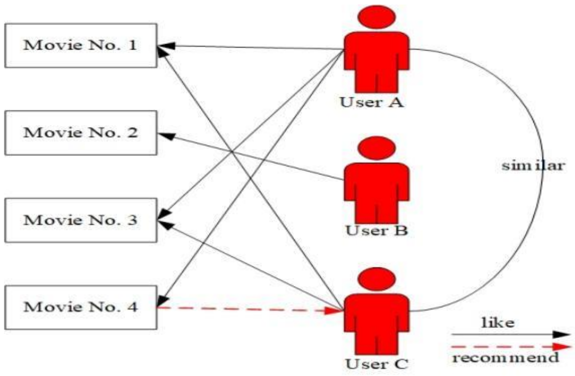

Fig.1. User-based CFA

2.2. Project-based CFA

This algorithm recommends the user's previous favorite items, analyzes the previous behavior records, calculates the similarity, and recommends the items to the user from high to low according to the previous preference for items in Fig.2.

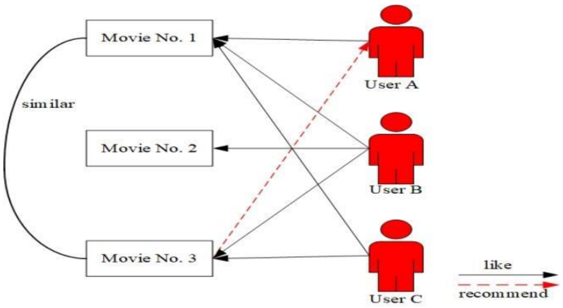

Fig. 2. Project-based CFA

2.3. Model-based CFA

This algorithm uses machine learning idea modeling to solve problems. A model is defined to describe the relationship between users and items, and the model parameters are obtained by optimizing the process. The correlation algorithm of machine learning is applied to the traditional CFA to simulate and train the historical behavior data of users. Get the relationship between users and projects, predict the data by model [12], as shown in Fig.3.

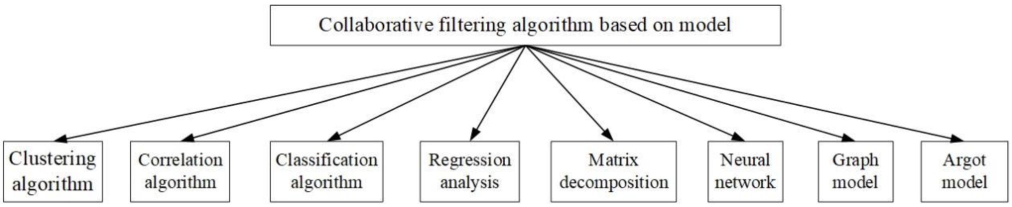

Fig. 3. CFA classification based on model

3. Traditional user-based CFA

3.1. Basic principles of the CFA

CFA finds a group of users with similar interests, and then recommends content to target users that they are interested in but haven't been exposed to. The flow chart of user-based CFA is shown in Fig.4.

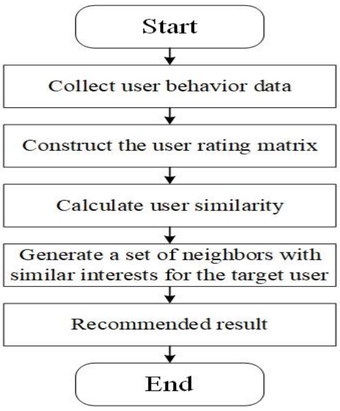
Start
Collect user behavior data
Construct the user rating matrix
Calculate user similarity
Generate a set of neighbors with similar interests for the target user
Recommended result
End

Fig.4. Flowchart of user-based CFA

3.2. Calculation of user similarity

In CFA, similarity calculation is always necessary. Similarity is used to measure whether items are similar to each other and whether the current user is similar to the previous user. There are many calculation methods for similarity, such as Jacquard similarity, cosine similarity, modified cosine similarity, Euclidean distance, Pearson similarity, etc. [13].

3.2.1 Jacquard similarity coefficient

The Jacquard similarity coefficient is specially used to calculate the similarity between finite sets. Generally speaking, Jacquard similarity coefficient of two groups indicates the degree of similarity between two groups. The calculation formula of Jacquard similarity coefficient is shown in Formula (1).

$$S_{u,v} = \frac{|N_{(u)} \cap N_{(v)}|}{|N_{(u)} \cup N_{(v)}|} \tag{1}$$

Where, $S_{u,v}$ represents the similarity between u and v, $N_{(u)}$ denotes the set that u interacts with, and $N_{(v)}$ denotes the set that v interacts with

3.2.2 Cosine similarity

Cosine similarity is used to measure the size of the vector angle between vector u and v, and show how similar u and v are. If the user vector is replaced, the similarity can be obtained in the same way, as shown in Formula (2).

$$S_{u,v} = \frac{V_{(u)} \cdot V_{(v)}}{\sqrt{|V_{(u)}||N_{(v)}|}} \tag{2}$$

Where, $V_{(u)}$indicates the score vector of u to the item, and $V_{(v)}$ denotes the score vector of v to the item.

3.2.3 Corrected cosine similarity

In Equation (2), the calculation of cosine similarity does not involve user scoring scale, that is, different users may give different scores to the same item according to their preferences or habits. Therefore, the above formula (2) needs to be modified and adjusted to take users' scoring preferences or habits into account [14], the modified cosine similarity can be obtained in Formula (3).

$$S_{u,v} = \frac{\sum_{c \in Iu,v}(R_{u,c}-\overline{R_u})(R_{v,c}-\overline{R_v})}{\sqrt{\sum_{c \in Iu}(R_{u,c}-\overline{R_u})^2}\sqrt{\sum_{c \in Iv}(R_{u,c}-\overline{R_v})^2}} \tag{3}$$

Where, Iu, v shows the collection of items that u and v have interacted with, $R_{u,c}$represents the score of u on item c, $R_{v,c}$indicates the score of v on item c, $\overline{R_u}$ *and* $\overline{R_v}$shows the average score of u and v on all interactive items, respectively. Iu and Iv shows the collection of items that u and v interacts with, respectively.

3.2.4 Pearson correlation coefficient

Pearson correlation coefficient is mainly obtained by calculating the quotient of covariance and standard deviation to obtain the similarity between two variables[15], Pearson correlation coefficient can be obtained in Formula (4).

$$S_{u,v} = \frac{\sum_{c \in Iu,v}(R_{u,c}-\overline{R_{uI}})(R_{v,c}-\overline{R_{vI}})}{\sqrt{\sum_{c \in Iu}(R_{u,c}-\overline{R_{uI}})^2}\sqrt{\sum_{c \in Iv}(R_{u,c}-\overline{R_{vI}})^2}} \tag{4}$$

Where, $\mathrm{R}_{\mathrm{u,c}}$ and $\mathrm{R}_{\mathrm{v,c}}$ show the score of u and v on item c, respectively. $\overline{\mathrm{R}_{\mathrm{uI}}}$ and $\overline{\mathrm{R}_{\mathrm{vI}}}$ indicate the average score of the items that u and user v have overrated together. Iu and Iv denote the set of items u and v interacted with, respectively.

4. Improved CFA based on user

Traditional CFA usually have problems such as data sparsity, cold start and real-time scalability [16]. This paper proposes to add the penalty term and time factor of popular items to the improved cosine similarity when calculating the user similarity, so as to optimize the calculation of the user similarity in order to avoid the influence of these problems [17].

4.1. Add a penalty item to the calculation of user similarity

In the recommendation system, the frequency of popular movies is too high, which will lead to a relatively simple recommendation result of system. Users see more popular movies than non-popular movies in the movie recommendation system, which does not well reflect the needs of users. For example, if two users have both seen Journey to the West, it cannot be shown that they have the same interests, most people have seen Journey to the West. However, if two users have both seen Psycho, it can be believed that they have similar interests and hobbies and like thriller movies to some extent [18]. To optimize the recommendation accuracy and reduce the influence of popular movies on user similarity calculation, IUCFAMR adds a penalty item to the calculation of the improved cosine similarity, where the formula of the penalty item is shown in Formula (5).

$$\frac{1}{\lg(1+|N(i)|)} \tag{5}$$

Where, $N(i)$ is the set of users who have behaved toward i-*th* movie. The improved cosine similarity after adding penalty term is shown in Formula (6).

$$S_{u,v}=\frac{\sum_{c\in Iu,v}\left(R_{u,c}-\overline{R_u}\right)\left(R_{v,c}-\overline{R_v}\right)\frac{1}{\lg(1+|N(i)|)}}{\sqrt{\sum_{c\in Iu}\left(R_{u,c}-\overline{R_u}\right)^2}\sqrt{\sum_{c\in Iv}\left(R_{u,c}-\overline{R_v}\right)^2}} \tag{6}$$

4.2. Add time factor to CFA

In the movie recommendation system, users' interest in movies will not stay the same, and their interests in different movies will change with time. To optimize the recommendation quality and real-time performance, time factor is added into the IUCFAMR algorithm, so as to better recommend movies that meet users' interests to users. The recent behavior of users is the closest reflection of users' current interests [19], the time decay function is added to the calculation of the corrected cosine similarity in IUCFAMR. The publicity of the time decay function is shown in Formula (7).

$$f(|t_{ui}-t_{vi}|)=\frac{1}{1+\alpha|t_{ui}-t_{vi}|} \tag{7}$$

Where, α is the time attenuation factor, $t_{ui}$and t_{vi} represents the time when u and v generate behavior on the item i, respectively. Adding the time attenuation function into the improved cosine similarity is shown in Formula (8).

$$S_{u,v}=\frac{\sum_{c\in Iu,v}\left(R_{u,c}-\overline{R_u}\right)\left(R_{v,c}-\overline{R_v}\right)f(|t_{ui}-t_{vi}|)}{\sqrt{\sum_{c\in Iu}\left(R_{u,c}-\overline{R_u}\right)^2}\sqrt{\sum_{c\in Iv}\left(R_{u,c}-\overline{R_v}\right)^2}} \tag{8}$$

To sum up the above, the IUCFAMR will fuse the penalty term and the time factor of the context of the popular movie and add the penalty term and time factor into the IUCFAMR formula, the calculation formula is shown in Formula 9.

$$S_{u,v}=\frac{\sum_{c\in Iu,v}\left(R_{u,c}-\overline{R_u}\right)\left(R_{v,c}-\overline{R_v}\right)f(|t_{ui}-t_{vi}|)\frac{1}{\lg(1+|N(i)|)}}{\sqrt{\sum_{c\in Iu}\left(R_{u,c}-\overline{R_u}\right)^2}\sqrt{\sum_{c\in Iv}\left(R_{u,c}-\overline{R_v}\right)^2}} \tag{9}$$

4.3. Core ideas of the improved CFA

The core of user-based CFA is to recommend users to other users with similar interests, find a set of users with similar interests to the target users, and then recommend items to users according to their purchase behaviors[20]. When calculating user similarity, time factor and penalty term are added into the improved cosine similarity in order to reduce the influence of popular movies on user recommendation and recommend users' favorite movies as accurately as possible, while taking into account that people's interests in the situation will change with time.

4.4 The algorithm flow chart of the IUCFAMR algorithm

The IUCFAMR flow is shown in Fig.5. and it is described as **IUCFAMR**.

Algorithm Name: **IUCFAMR**

```
Begin
   def __init__(self, datafile); // initializes the function
 self.train, self.test, self.max_data = self.loadData(); //max_data represents the
                                                          //maximum score
   def loadData(self); // Loads the data set and splits it into a training set and a test set
   count = dict()  ;
   user_eval_item_count = dict();
for i, users in item_eval_by_users.items()
```

```
         userSim = dict() ;
for u, related_users in count.items()
        userSim.setdefault(u, {}) ;
        for v, cuv in related_users.items()
        def UserSimilarityBest(self) ;
        count[u][v] += 1 / ( 1+ self.alpha * abs(self.train[u][i]["time"]- self.train[v][i]
["time"] ) / (24*60*60) ) \ * 1 / math.log(1 + len(users));
        def recommend(self, user, k=20, nitems=20);
        End
```

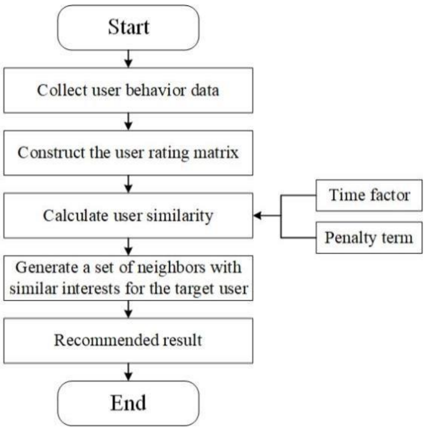
Start
Collect user behavior data
Construct the user rating matrix
Calculate user similarity
Time factor
Penalty term
Generate a set of neighbors with similar interests for the target user
Recommended result
End

Fig.5. Improved user-based CFA

5. Experimental results and analysis

5.1 Experimental environment and platform

The experimental platform adopted the following configurations: AMD Ryzen7 5800H, Radeon Graphics 3.20GHz, windows10 operating system, 16.0GB memory, 1T hard disk and PyCharm Community Edition 2021.

5.2 Data Set

Table 1.

Data set structure

Table name	Fields	Instructions
movies	movieId	Movie number
	title	Movie title
	genres	Genres of movies
ratings	userId	UserID
	movieId	Movie number
	rating	Rating (0.5-5)
	timestamp	Time stamp
	userId	UserID
	movieId	Movie number

tags	tag	Tags
	timestamp	Time stamp
links	movieId	Movie number
	imdbId	Number in imdb
	tmdbId	Number in tmdb

This experiment uses the ml-latest-small data-set from the MovieLens dataset commonly used by movie recommendation systems, which is from the GroupLens research group at the University of Minnesota [21]. The data-set contains 610 users, 9,742 movies and 100,837 rating records, which is shown in Table 1.

5.3 Evaluation Index

The evaluation indexes mainly include Recall rate and Precision rate to test the recommended accuracy. The larger the recall rate and precision rate, the more accurate the classification will be[22]. The recall rate mainly refers to the proportion of the items that the user likes in all the items that the user really likes. When predicting whether the user likes them, the recall rate formula is shown as (10). The accuracy rate mainly refers to the proportion of the items that the user likes in all the items that the user really likes. When predicting whether the user likes them, the accuracy rate formula is shown as formula (11).

$$Recall = \frac{\sum_{u\in U}|L(u)\cap S(u)|}{\sum_{u\in U}|S(u)|} \tag{10}$$

$$Precision = \frac{\sum_{u\in U}|L(u)\cap S(u)|}{\sum_{u\in U}|L(u)|} \tag{11}$$

Where, $L(u)$is the collection of items recommended to u, $S(u)$is the collection of items included in the recommendation list and selected by u, and u is the collection of all users.

5.4 Experimental results and analysis

The number of users' nearest neighbors K is set to 5, 10, 15, 20, 25, 25, 30, 35, 40, 45, 50, respectively and the number of movie recommendations as 20. It can be seen from Fig.6 that Precision and Recall change smoothly in the the traditional CFA. Therefore, the selection of K value in the experiment has little influence on the experimental results. Fig.6 shows the Precision and Recall values under different K values.

The user's nearest neighbor K is set as 20, and the movie recommendation N is set as 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, respectively. In the calculation of user similarity, the IUCFAMR comprehensively considers the influence of popular items and time context in calculation of user similarity. The penalty term and time factor for popular items are added to the improved cosine similarity formula. It can be seen from Fig.7 and 8. That, compared with the traditional CFA, the Precision values and Recall values of the IUCFAMR have been improved to some extent, which can solve the problems of data sparsity and real-time scalability.

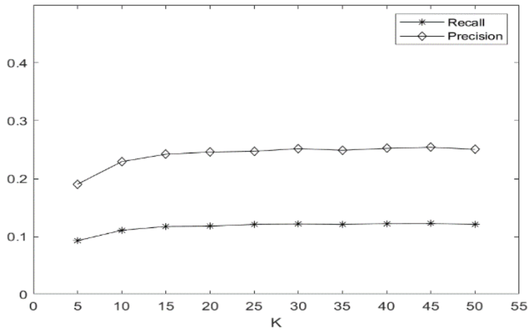

Fig.6. Recall and Precision values under different K values

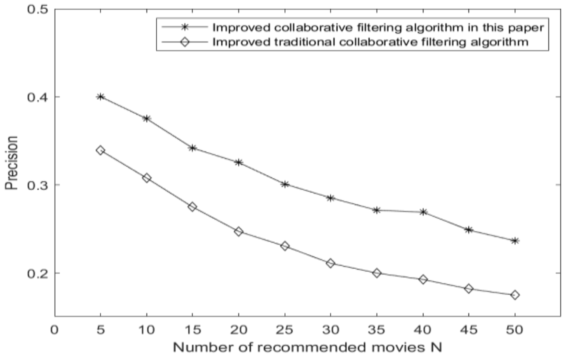

Fig.7. Comparison of Precision values

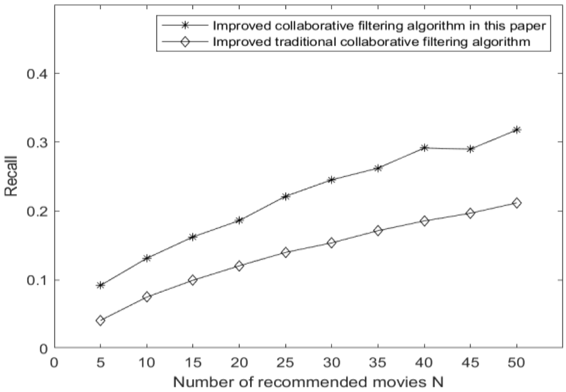

Fig.8. Comparison of Recall values

6. Conclusions

Through systematic investigation, it is found that the traditional CFA has insufficient data sparsity and real-time scalability, which will lead to the recommendation of popular movies but not the user's favorite movies to users in the movie recommendation system, affecting users' choice of non-popular movies. The IUCFAMR is proposed to add a penalty term to solve this problem to optimize the recommendation effect. On the other hand, the user's hobby will gradually change with the change of time. IUCFAMR also adds time factor to alleviate this problem. By adding penalty term and time factor, IUCFAMR solves the traditional problem of insufficient expansion of data sparsity and real-time.

The calculation of user similarity is improved in the IUCFAMR by integrating the time factor and the penalty item for popular projects. Systematic experiments can fully prove that the IUCFAMR can play a better recommendation effect in the movie portal system with the change of user behavior, however, in order to better judge user behavior and further improve the recommendation effect, the research team will continue to make efforts to carry out optimization research.

Acknowledgement

The work was supported in part by Young Talents of Anhui Province under Grant gxyq2019175, in part by Quality project of Anhui Province with Grant number 2018mooc061, 2020mooc386, 2020zdxsjg260, 2020jyxm1423, 2021sx117, 2021jxtd215 and 2021jyxm1121, in part by start-up project of Fuyang Normal University under grant 2019kyqd0018 and in part by Anhui Province university key research project under grant KJ2018A0669 and 2022AH052820.

REFERENCES

[1] G. Tana. “Recommendation Algorithm for Potential Candidates in Human Resources Based on the Integration of Dynamic Decision Diagram”. 2019 International Conference on Smart Grid and Electrical Automation (ICSGEA), 10-11 August 2019, Xiangtan, China.

[2] Q.C. Yu, M.Q. Zhao, Y.T. Luo. "CFA based on Optimal Weight". International Conference on Computer Engineering and Artificial Intelligence (ICCEAI), 22-24 July 2022, Shijiazhuang, China.

[3] R.G. Lumauag, A. M. Sison, R. P. Medina. "An Enhanced Recommendation Algorithm Based on Modified User-Based CFA". IEEE 4th International Conference on Computer and Communication Systems (ICCCS), 23-25 February 2019, Singapore.

[4] İclal ÖZCAN and M. ÇELİK."Developing Recommendation System Using Genetic Algorithm Based Alternative Least Squares", International Conference on Artificial Intelligence and Data Processing (IDAP), 28-30 September 2018, Malatya, Turkey.

[5] P.F. Zhang. "Design and implementation of wechat Ordering recommendation system based on CFA". Changchun: Jilin University, 06, 2022.

[6] B.B. Qian. "Design and Implementation of Music Recommendation System based on CFA". Fuyang: Fuyang Normal University, 06, 2022.

[7] Lee Cheong Rok. "An improved similarity measure for CFA recommendation system". International Journal of Knowledge-based and Intelligent Engineering Systems. 2022, 26(03):137-147.

[8] Y.P. Du. "Research on Personalized Book Recommendation Based on Improved Similarity Calculation and Data Filling CFA". Computational Intelligence and Neuroscience.2022, 2022(1900209):1-15.

[9] J. Gourav. "TD-DNN: A Time Decay-Based Deep Neural Network for Recommendation System. Applied". Sciences.2022, 12(13):6398-6419.

[10] D. Z. Ulian et al., "Exploring the effects of different Clustering Methods on a News Recommender System". Expert Systems with Applications. 2021, 183(115341):1-14.

[11] D.S. Li., "Recommendation System Frontier and Practice". Beijing: Publishing House of Electronics Industry, 05.2022.

[12] Z. Wang. "Deep Learning Recommendation System". Beijing: Publishing House of Electronics Industry, 07, 2020.

[13] Y.T. Gao. "Practical Recommendation System Development". Beijing: Publishing House of Electronics Industry, 09, 2019.

[14] J.K. Wu. "Research on CFA Based on Improved Similarity". Computer Technology and Development, 2022, 32(04):39-43.

[15] L.S. Cui. "Research on CFA Based on User Characteristics and item type interest". Zhengzhou: Henan University of Economics and Law, 07, 2022.

[16] W. Triyanna. "Recommendation Algorithm Using Clustering-Based UPCSim (CB-UPCSim)". Computers.2021, 10(10):123-139.

[17] A. Sajad et al., "Alleviating data sparsity problem in time-aware recommender systems using a reliable rating profile enrichment approach". Expert Systems with Applications. 2022, 187(115849):1-15.

[18] Z.L. Wang. "Improved CFA library recommendation Model integrating time Context". Science and Technology Wind, 2021, (03):131-132.

[19] T.S. Qu. "Research on Personalized music Hybrid Recommendation Algorithm based on Time Context Information". Jinzhou: Bohai University, 07, 2021.

[20] X. Zhou. "Collaborative filtering book recommendation Algorithm with penalty factor". Jiangsu Science and Technology Information. 2022, 39(17):76-80.

[21] Z.F. Zhang. "Research on Personalized Movie Recommendation Algorithm Based on Time Context". Journal of Hebei Software Vocational and Technical College, 2020, 22(04):5-8.

[22] H. Fang. "CFA to Improve Item Similarity Calculation". Software Guide. 2021, 20(09):88-92.