ITC 1/54

Information Technology and Control
Vol. 54 / No. 1/ 2025
pp. 64-83
DOI 10.5755/j01.itc.54.1.37088

Personalized Intelligent Recommendation Model for Educational Games Based on Data Mining

Received 2024/04/26 | Accepted after revision 2024/08/13

HOW TO CITE: Yang, M., Li, D. (2025). Personalized Intelligent Recommendation Model for Educational Games Based on Data Mining. *Information Technology and Control,* 54(1), 64-83. https://doi.org/10.5755/j01.itc.54.1.37088

Personalized Intelligent Recommendation Model for Educational Games Based on Data Mining

Min Yang

School of Literature and Media, Xi'an FANYI University, Xi'an 710105, China

Dandan Li

Dept of Education and Culture Contents Development, Woosuk University, Wanju-gun 55338, Republic of Korea

Corresponding author: DandanLi0731@outlook.com

To solve the problem that educational game recommendation is not effective due to data sparsity, this paper proposes a collaborative filtering recommendation (CFR) algorithm that integrates covering rough granularity layer clustering (CRGLC) and K-means clustering. On the basis of the K-means clustering CFR model, the paper introduces the granularity calculation and constructs the covering rough granularity space according to the user's comprehensive score and game type. The performance test results showed that the accuracy and F1 score of the improved algorithm are 0.880 and 0.826 respectively, which are higher than those of the comparison algorithm. In the actual application performance test, it is found that the clustering performance of the model is good because the difference within the cluster is small and the difference outside the cluster is large. Compared with the known better recommendation algorithms, the MAE and RMS errors of its score prediction are the lowest. The above results show that the algorithm has higher accuracy in educational game recommendation. In general, the innovation of the algorithm lies in the fusion of CRGLC and K-means clustering, and the introduction of granular computation to deal with data sparsity and improve the recommendation accuracy. This research has some practical value to solve the problem of sparse data in educational game recommendation.

KEYWORDS: Data mining; Educational games; Recommendation algorithm; K-means clustering; Collaborative filtering; Covering rough granular layer clustering.

1. Introduction

With the rapid development of information technology, games have become an important way of leisure and entertainment for people [34]. In China, the number of online game users has exceeded 522 million, among which educational games have attracted wide attention due to their integration of educational and entertainment elements. However, in the face of massive game information, how to accurately recommend suitable educational games to users has become a challenge [35]. The recommendation system excavates potential interests by analyzing user behaviors and attributes, and then makes personalized recommendations. However, data sparsity affects the accuracy and diversity of traditional recommendation algorithms [26]. The novel feature of this study is the integration of CRGLC and K-means clustering to propose an innovative CFR algorithm. This combination not only improves the global search ability, but also effectively reduces the impact of data sparsity, thus significantly improving the accuracy and diversity of recommendation algorithms. Compared with existing studies, the proposed method captures users' interests more accurately by refining user proximity, which is a unique contribution to the integration of granular computing (GC) and clustering technology in related fields [19]. The research aims to verify the effectiveness of this method through experiment, provide new ideas for improving the recommendation algorithm of educational games, better meet the needs of users, and promote the application of educational games in the field of education.

The research mainly contains five parts. The first part introduces the current development status of data mining (DM) technology and the research progress of recommendation algorithms, and proposes an improved algorithm system to address the shortcomings of recommendation systems. The second part explains in detail the application methods and processes of the algorithm in the recommendation model from two aspects: the construction and improvement of the educational game recommendation (EGR) algorithm. The third part conducts experimental verification and analysis on the proposed algorithm model, and tests the performance of the recommendation model. The fourth part is the in-depth analysis and discussion of the research results. The fifth part summarizes the results of this study, analyzes its shortcomings, and proposes directions for improvement.

2. Related Works

In today's big data era, DM technology has become the key to analyze massive information, and it has excellent performance in many fields such as medical care and business. For example, the rehabilitation training prediction system developed by Tuah team, which integrates DM technology and gamification concept, has customized personalized rehabilitation training for stroke patients and significantly improved the rehabilitation effect [27]. In the aspect of clustering algorithm, K-means algorithm is widely used, but there are many challenges. By improving the artificial bee colony algorithm, combining the globally guided fitting function and the new position update formula, the Yao team not only improved the optimization efficiency, but also solved the problem of local optimal solution, making the K-means clustering effect better [33]. At the same time, Zhang et al. proposed a new method combining secure multi-party computing and differential privacy technology to realize the dual protection of data privacy and effective clustering in view of the privacy protection defects of K-means [36]. In addition, Liu et al. also enhanced the stability of K-means algorithm by optimizing the selection of initial clustering centers [18]. In addition, there are many scholars on the CRGLC algorithm research. For example, Jain and Som proposed a novel robust model combining intuitive fuzzy set β coverage and multi-granularity rough sets to address the limitations of fuzzy and intuitional fuzzy β coverage in decision applications. Experimental results showed that the proposed model has significant advantages over traditional methods [11]. Deboeuf and Fall proposed a rheometer parameter optimization method based on CRGLC algorithm to solve the problem of non-three-phase rheological properties in multiphase flows, and conducted an empirical analysis of the method. The results showed that the rheometer parameters obtained by this method can effectively improve the overall performance of the rheometer and are highly practical [5].

With the promotion of recommendation system, recommendation system has gradually become a key technology for major platforms to improve user experience. Many scholars at home and abroad have studied the performance and privacy protection of recommendation systems optimized by clustering algorithm

Table 1

Comparison between the advantages and disadvantages of the proposed research method and the related literature method

Literature/ author	Technology/method	Advantages	Disadvantages	Research proposed methods to compensate
Tuah et al. [27]	A rehabilitation system that integrates DM and gamification concepts	Personalized rehabilitation training, rehabilitation effect is better	Limited to specific areas	The research method can be applied to a wider range of fields, such as EGR
Yao et al. [33]	Improving artificial bee colony algorithm to optimize K-means	The optimization efficiency and clustering effect are good	High complexity	The research method further improves the global search ability and recommendation accuracy by integrating CRGS thought
Zhang et al. [36]	K-means combining secure multi-party computation and differential privacy	Dual protection of data privacy and effective clustering	High computational overhead	The research method pays attention to improving the accuracy of recommendation while protecting privacy
Liu et al. [18]	Optimize the K-means of initial cluster center selection	K-means algorithm has strong stability	Sensitive to initial cluster center selection	Methods PCA was used to improve the scoring system and reduce the dependence on the initial conditions
Chen et al. [3]	A game-based evolutionary clustering recommendation method	The personalized recommendation and user engagement are strong	Limited by the game field	The research method is not limited to games, but can be applied to a wider range of recommendation scenarios
Shafiq et al. [24]	A recommended approach combining federated learning and game theory	Provide accurate recommendations while protecting user privacy	Requires complex computation and communication overhead	The research method simplifies computation and communication complexity while ensuring privacy
Liu et al. [20]	Triple cross-domain CFR	Improve the accuracy and variety of recommendations	Faced with the challenge of data integration	The research method improves the ability of global search and mining of hidden information by integrating CRGS thought
Demirkiran et al. [6]	The rough set theory is used for multi-criteria collaborative filtering	It can effectively deal with the uncertainty and ambiguity in the recommendation process	High computational complexity	The research method pays attention to the calculation efficiency while improving the accuracy of recommendation

and federated learning. For example, in order to improve the recommendation effect of games, Chen et al. proposed an evolutionary clustering method based on games, which combines historical information to make personal recommendations, and introduces the concept of gamification to improve the personalized recommendation and user engagement [3]. Shafiq et al. proposed a method combining federated learning and game theory to address the low security of online consumer recommendation, aiming at providing personalized security recommendation for consumer electronic devices in 5G and Internet of Things environments. The results showed that this method can effectively protect user privacy and provide a new trend of accurate recommendation [24]. To solve the problem of data sparsity in the recommendation process, Liu et al. proposed triple cross-domain CFR, and the results showed that this method can effectively enhance the accuracy and diversity of recommendation by integrating information from different fields [20]. To improve the accuracy of e-commerce recommendation, Demirkiran et al used rough set theory to carry out multi-criteria collaborative filtering, and the results showed that this method provided a new perspective for dealing with uncertainty and fuzziness in the process of e-commerce recommendation [6]. In addition, Huang et al. discussed the design of federated learning incentive mechanism for recommendation systems in mobile edge computing environment, aiming to solve the balance between data privacy and computational efficiency [10].

Based on the above related literature, the advantages and disadvantages of the proposed research method are compared, and the comparison results are shown in Table 1. From Table 1, the current research has shortcomings such as high complexity, limited recommendation fields and sensitive parameters. However, the proposed method makes up for the above shortcomings of the existing research by integrating the idea of covering rough granular space (CRGS) into the K-means clustering CFR process and improving the user comprehensive scoring system by combining the principal component analysis (PCA) method. Among them, a method is proposed to enhance the model's mining of hidden information, improve the global search ability, and improve the recommendation accuracy rate. At the same time, this method also enhances the data processing effect of uncertainty and fuzziness by incorporating the idea of rough set.

3. Design of an EGR Model Based on Cluster Analysis

In recent years, the application of CFR algorithm has been improved to solve the problem of data cold start. The CRGS idea in GC is innovatively combined with K-means clustering, and an improved CFR algorithm is proposed. The algorithm first divides the user's interest space into several coarse grains, refines the user's proximity range, and then uses K-means clustering algorithm to cluster the user data in the coarse-grained layer, and classifies the users with similar interests into the same layer. This increases the accuracy and variety of recommendations, which is of great value in educating game users about loyalty and engagement.

3.1. EGR Algorithm Based on K-means Clustering CFA

Due to the fact that the EGR model faces different types of users and their preferences for educational games vary, it is necessary to construct a user tag system. Tagging user browsing behavior can enable the model to quickly obtain relevant information such as user social attributes, personal behavior, and interest preferences, to achieve real-time intelligent recommendations. Tags describe the features exhibited by a certain type of person, and by combining all features, a tag system with specific attributes can be formed. The original data of the tag can be obtained from the public website of the educational game development company. According to the user browsing behavior data, this research divides the original dataset into three types, namely, the basic information, browsing situation, and subjective perception information of the game user, and constructs the tag system as shown in Figure 1.

The PCA method is applied to assign and lessen the weights of user tags. Using user tags as dimensions, from Figure 1, the dimension is 7. It assumes the user set is $U=\{u_1,u_2,\cdots,u_m\}$, and constructs a matrix with 7 rows and m columns, and the expression for the feature $\mathbf{Z}'$ of the 7-dimensional data sample is shown in Equation (1).

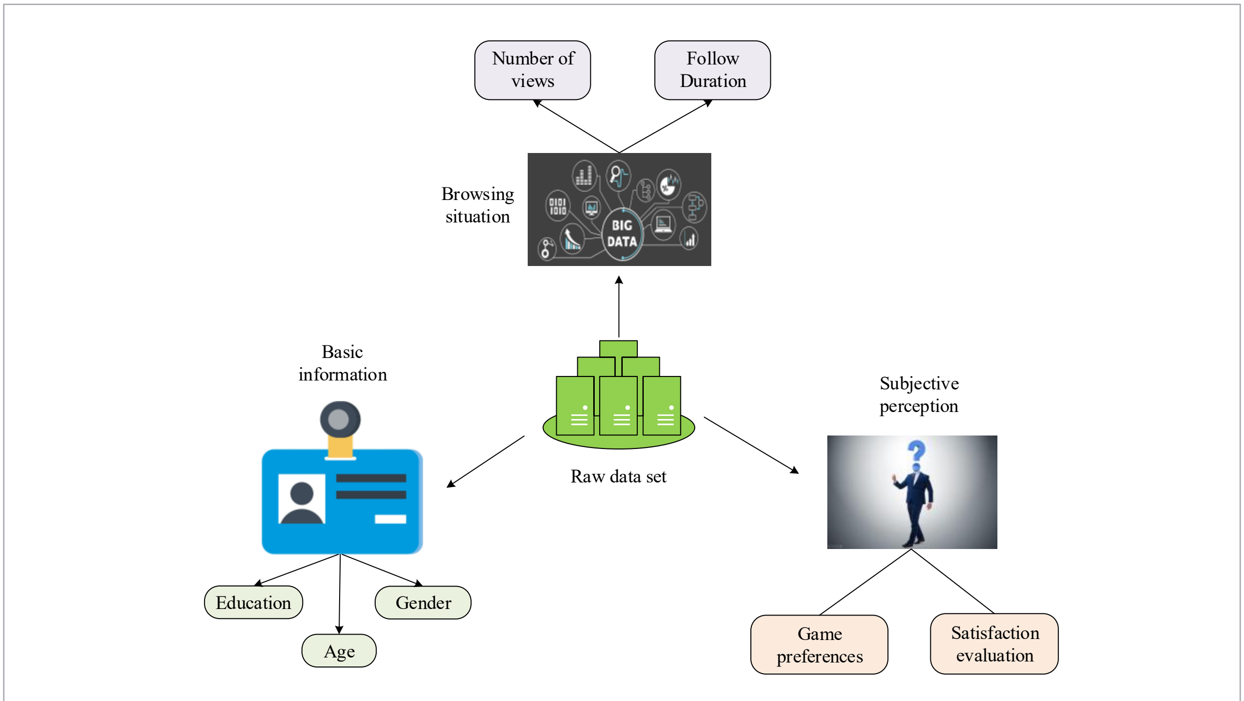
$$\mathbf{Z}'=\begin{bmatrix} z'_{11} & z'_{12} & \cdots & z'_{1m} \\ z'_{21} & z'_{22} & \cdots & z'_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ z'_{71} & z'_{72} & \cdots & z'_{7m} \end{bmatrix} \tag{1}$$

It centralizes each row of data according to Equation (2) and organizes it into 7 row vectors. The 7 original variables are represented by $Z_1, Z_2, \cdots, Z_7$.

$$\begin{cases} z_{wu} = z'_{wu} - \dfrac{\sum_{x=1}^{m} z'_{wx}}{m} \\ \mathbf{Z}=\begin{bmatrix} z_{11} & z_{12} & \cdots & z_{1m} \\ z_{21} & z_{22} & \cdots & z_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ z_{71} & z_{72} & \cdots & z_{7m} \end{bmatrix} = \begin{bmatrix} Z_1 \\ Z_2 \\ \vdots \\ Z_7 \end{bmatrix} \end{cases} \tag{2}$$

In Equation (2), u and w represent the amount of users and dimensions, respectively. It calculates the covariance matrix $\mathbf{W}$ of the sample, decomposes the covariance matrix, and calculates the eigenvalues and eigenvectors. The expression for

Figure 1
Label System Based on Browsing Behavior Data of Educational Game Users



calculating covariance W is shown in Equation (3).

$$W = \frac{ZZ^{\mathrm{T}}}{m} \tag{3}$$

To convert 7-dimensional data into 1-dimensional data and reduce information loss caused by dimensional changes, it calculates the ratio δ_{wu} of each label feature vector to the eigenvalues of the root sign, and combines the variance contribution rate φ of the principal component feature values to calculate the weight of the label, as shown in Equation (4).

$$\gamma_w = \frac{\sum_{u=1}^{m} \varphi_w \delta_{wu}}{\sum_{o=1}^{w} \varphi_o} \tag{4}$$

It multiplies and adds the weights with the corresponding tag features to obtain a user's comprehensive rating for a game. Then, clustering algorithms are used to analyze and calculate the data. Common clustering algorithms include aggregate clustering, Gaussian mixture, Mean Shift, K-means, etc. Most algorithms measure dense observation areas using similarity and distance between samples in the feature space. The K-means is a typical algorithm in DM, which is widely applied in clustering problem analysis due to its advantages of simplicity, ease of operation, and ability to handle large datasets. Due to the poor recommendation performance of the K-means clustering algorithm when facing data sparsity issues, this study optimizes the selecting clustering centers [18]. To ensure the representativeness of cluster centers, a statistic and computation-based method is adopted to select cluster centers. Firstly, the method of mean value is used to select the first center point. If the educational game is represented by the set R, and the user's rating of the game is expressed as r_{ui}, a matrix R for different user ratings of the game can be constructed. It takes the average score of m users on n games as the first clustering center K_1 to avoid the subsequent inability to accurately find neighbors caused by randomly selecting the clustering center. The calculation is shown in Equation (5).

$$\begin{cases} K_1 = \left\{k_{i_1}^1, k_{i_2}^1, \cdots, k_{i_n}^1\right\} \\ k_{i_1}^1 = \dfrac{1}{m}\sum_{x=1}^{m} r_{u_x i_1} \end{cases} \tag{5}$$

In Equation (5), $k_{i_n}^1$ denotes the user's average rating of the game i_n. To further enhance the rationality of the clustering center, standard deviation is introduced to quantify the degree of dispersion of users' ratings of games. By calculating the standard deviation of each user rating from the first cluster center, it can better understand the distribution of the data and adjust the selection of subsequent cluster centers accordingly. In the selection of subsequent clustering centers, the study follows the principle of maximum and minimum distance. For each new cluster center, the distance between it and the existing cluster center is calculated, and the point farthest from the previous cluster centers is selected as the new cluster center. It calculates the distance between all user ratings and the first cluster center according to Equation (6), and uses the point corresponding to the maximum distance as the second cluster center K_2.

$$\begin{cases} dist(u,v) = \sqrt{\sum_{y=1}^{n}\left(\dfrac{r_{ui_y} - r_{vi_y}}{\sigma_{i_y}}\right)^2} \\ \sigma_{i_y} = \sqrt{\dfrac{1}{m}\sum_{x=1}^{m}(r_{u_x i_y} - \mu)^2} \end{cases} \tag{6}$$

In Equation (6), μ expresses the standard deviation of user ratings for the game. μ represents the average user rating of the game. It filters all obtained cluster centers according to the principle of maximum and minimum distance. It assumes the jth cluster center is K_j, calculates the distance from K_3 to the first two cluster centers K_1 and K_2 when $j=3$, and finds the maximum value DL_3 of the minimum distance between K_3 and the first two cluster centers, as shown in Equation (7).

$$DL_3 = \max\{\min[d(K_3,K_1), d(K_3,K_2)]\} \tag{7}$$

In Equation (7), d indicates the distance. If there are already $j-1$ cluster centers, it needs to calculate the distance between the jth cluster center as K_j and the first d cluster centers, and find the maximum value DL_j of the minimum distance, as shown in Equation (8).

$$DL_j = \max\{\min[d(K_j,K_1), d(K_j,K_2), \cdots, d(K_j,K_{j-1})]\} \tag{8}$$

It uses the minimum distance as the threshold ε to assign clusters to sample points, iterates according to Equation (9), and ends the selection of cluster centers when the conditions are met. The threshold value of the algorithm model in this study is 2.

$$\max[d(K_{j+1}, K_j)] < \varepsilon \tag{9}$$

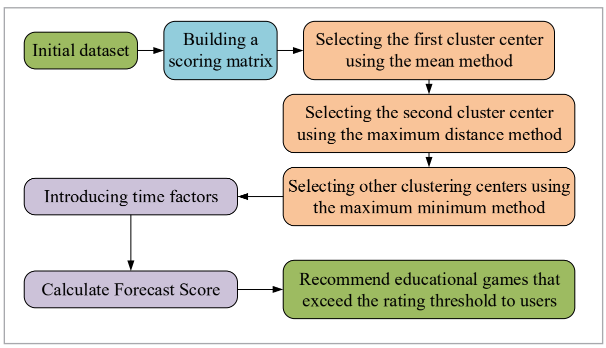
Since users' preferences for games change over time, the study optimizes the time factor and sets a distance threshold to select other clustering centers. By calculating the distance of each point from the first cluster center, the farthest point is selected as the second cluster center. This step is designed to ensure differentiation between clustering centers, allowing for more comprehensive coverage of the data space. To mitigate the impact of time, the time factor $f(t)$ is introduced and optimized in the calculation of the algorithm's prediction score p_{ui}. The calculation expression is shown in Equation (10).

$$\begin{cases} f(t) = \dfrac{y}{n\cdot(1+e^{-(t_i-t_0)})} \\ p_{ui} = \bar{r}_u + \dfrac{\sum_{v\in S(u,l)\cap N(i)} [sim(u,v)\times f(t)\times(r_{vi}-\bar{r}_v)]}{\sum_{v\in S(u,l)\cap N(i)} [|sim(u,v)|\times f(t)]} \end{cases} \tag{10}$$

In Equation (10), t_i and t_0 respectively mean the time when the user interacts with the game and the time when the user first plays the game. $sim(u,v)$ refers to the similarity of comprehensive labels between user u and v. $S(u,l)$ denotes the lth adjacent user. u stands for a collection of users who have interactive information with the game. y refers to the yth game [7]. Finally, the study also filters recommendations by setting a rating threshold, ensuring that only games with predicted scores above this threshold are recommended to the target audience, a step designed to improve the accuracy of recommendations and user satisfaction. Figure 2 shows the recommendation

Figure 2

Recommendation Process of K-means Clustering Algorithm



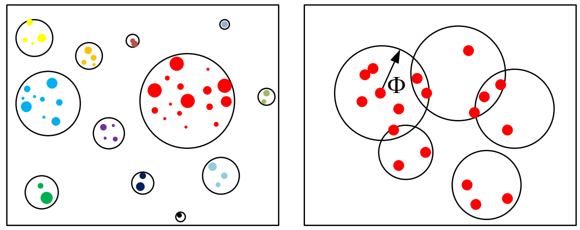
process of the improved K-means algorithm, and it shows the whole process from data preprocessing to cluster center selection to final recommendation.

3.2. Optimization Algorithm of CFR Model Integrating CRGLC

The key of CFA is to find the nearest neighbor, which can be realized through K-means clustering analysis. Although the K-means clustering can raise recommendation effectiveness to some extent, it still falls into local optima, resulting in unsatisfactory recommendation performance. For this reason, many researchers have improved the recommendation model. Davtalab et al. constructed a user social and geographic similarity model to incorporate user social relationships into the recommendation process, deeply mining user preferences, and improving model recommendation accuracy [4]. Song et al. combined user clustering algorithm with scoring preference Slope One algorithm to raise the accuracy of model scoring prediction. Due to the model that simply integrates user tags and game attributes into the recommendation process, it is not possible to dynamically adjust recommendation lists of different granularity and levels based on changes in user characteristics [25]. The simple use of K-means clustering algorithm can enhance the similarity of users by grouping users with similar interests into the same cluster. However, this approach sacrifices the personalization of recommendations because it focuses more on commonalities among users than differences. In order to improve the K-means clustering algorithm, GC is introduced on the basis of the K-means recommendation algorithm, and the influence of data sparsity is weakened by finding the local rough particle (LRP) set of game users. In addition, additional personalization factors, such as the user's personal characteristics, historical behavior, or real-time feedback, are introduced into the model to fine-tune the clustered recommendation results, ensuring that the recommendations are both targeted and differentiated. It assumes D is a finite non-empty domain with a subset of C, and satisfying $\bigcup C = D$, then C is the coverage of D, and (D,C) is the coverage approximation space [32]. The schematic diagram of particle division and coverage is displayed in Figure 3.

For user set $U=\{u_1,u_2,\cdots,u_m\}$, performing a reduction operation on C can obtain a covering rough par-

Figure 3
Schematic Diagram of Particle Division and Coverage



(a) Partitioning of particles (b) Coverage diagram

ticle set $reduct(C)=\{\underline{C}(u_1),\underline{C}(u_2),\cdots,\underline{C}(u_m)\}$. If there is $\forall u\in D$ in the coverage approximation space (D,C), then the minimum description $Md(u)$ of user u's coverage rough particle can be represented by Equation (11) [17].

$$Md(u)=\left\{A\in C \middle| u\in A\wedge(\forall B\in C\wedge u\in B\wedge B\subseteq A\Rightarrow A=B)\right\} \tag{11}$$

In Equation (11), both A and B are sets. Due to the R type set of educational games, the user's LRP set can be expressed as Equation (12).

$$\begin{cases} C=\{C_{i_1},C_{i_2},\cdots,C_{i_n}\} \\ C_i=\{Md_i(u_1),Md_i(u_2),\cdots,Md_i(u_m)\} \end{cases} \tag{12}$$

Here, the coverage of the minimum description of rough grains denotes the user's preference for game types, with $Md_i(u_m)$ indicating the subset C denoting the set of user preferences for each type of game. It assumes that X means the equivalence relation of D, $\forall Y\subseteq D$, the lower approximation set $\underline{C}(Y)$ and the upper approximation set $\underline{C}(Y)$ can be obtained according to Equation (13) [30].

$$\begin{cases} \underline{C}(Y)=\bigcup\left\{Md(u)\in D/X \middle| Md(u)\subseteq Y\right\} \\ \overline{C}(Y)=\bigcup\left\{Md(u)\in D/X \middle| Md(u)\bigcap Y\neq\varnothing\right\} \end{cases} \tag{13}$$

The upper and lower approximation sets separate the universe into positive domain $\underline{C}(Y)$, negative domain $\underline{C}(Y)$, and boundary domain $\overline{C}(Y)-\underline{C}(Y)$. The global coverage rough set of user u_m in the coverage approximation space is $reduct(C_{u_m})=\{\underline{C}_{i_1}(u_m),\underline{C}_{i_2}(u_m),\cdots,\underline{C}_{i_n}(u_m)\}$. If the global covering rough particle (GCRP) set meets the following conditions: (1) any GCRP cannot be obtained by the union operation of LRPs; (2) any GCRP cannot be replaced by the union operation result of LRPs. So, by performing union operations on the GCRP set, the CRGS can be obtained [16]. In this space, based on the weights obtained by PCA, it calculates the proportion of the number of games that interact with users in the total amount of games of the game type i, to obtain the user's preference degree q_{ui}. And the average preference degree $\overline{q}_{ui}$ of users for all game types is calculated as shown in Equation (14).

$$\begin{cases} q_{ui} = \dfrac{s_{ui}}{\sum\limits_{u=1}^{m} s_{ui}} \\ \overline{q}_u = \dfrac{q_{ui}}{\sum\limits_{i=1}^{n} q_{ui}} \end{cases}, \ (u = 1,2,\cdots,m) \ , \ (i = 1,2,\cdots,n) \tag{14}$$

In Equation (14), s_{ui} denotes the amount of games that interact with users. The degree of user preference is defined as the user's LRP set, which forms a mapping relationship between users and "particles" [2, 38]. It constructs a preference matrix $\mathbf{Q}$ as shown in Equation (15) based on the user's preference level.

$$\mathbf{Q} = \begin{bmatrix} q_{u_1 i_1} & q_{u_1 i_2} & \cdots & q_{u_1 i_n} \\ q_{u_2 i_1} & q_{u_2 i_2} & \cdots & q_{u_2 i_n} \\ \vdots & \vdots & \ddots & \vdots \\ q_{u_m i_1} & q_{u_m i_2} & \cdots & q_{u_m i_n} \end{bmatrix} \tag{15}$$

CFA is a recommendation algorithm based on near neighbor users. The core idea is that when a user has a demand, the algorithm recommends games that do not generate interactive information to other users who have similar interests [1]. The algorithm principle is shown in Figure 4.

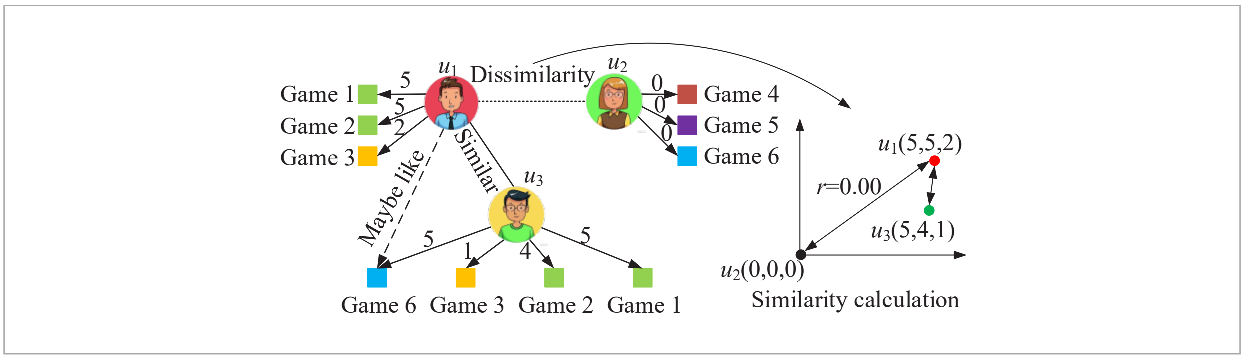
Euclidean distance, Jaccard, cosine, or Pearson correlation coefficients are commonly used to calculate similarity [28]. To comprehensively consider the similarity between users, this study combines the Pearson coefficient based on spatio-temporal features and the user preference level in matrix $\mathbf{Q}$ for calculation, as shown in Equation (16) [15].

$$sim(u,v) = \begin{cases} 1 \ , \ u = v \\ \dfrac{\sum\limits_{x\in(u,v)} (q_{u,x} - \overline{q}_u)(q_{v,x} - \overline{q}_v)}{\sqrt{\sum\limits_{x\in(u,v)} (q_{u,x} - \overline{q}_u)^2} \times \sqrt{\sum\limits_{x\in(u,v)} (q_{u,x} - \overline{q}_u)^2}}, \ u \neq v \end{cases} \tag{16}$$

It sets different threshold α_i based on the user's preference for game types, and uses this as a basis to find the user's nearest neighbor set in different game projects. It identifies users with similarity values greater than the threshold χ as close neighbors of the target user across all game types. It finds the LRP set of the user in the game type layer and sets the coverage coefficient Φ, which represents the degree of coverage to the user's nearest neighbors [39]. Assuming the target user is u_1, if there are two users u_a and u_b, and $a,b \neq 1$, it compares the similarity between u_1, u_a, and u_b under different game types i, and sets the smaller similarity value as the coverage coefficient Φ. Based on

Figure 4

Principle of CFA



Φ's coverage of the target user's proximity, the lower approximation set $\underline{C}_{i_1}(q_{u_1})$ in game type i is obtained. It calculates the minimum description $Md_i(q_{u_1})$ of the target user under each game type layer, and combines it to form the local covering rough particle set of u_1 under the coverage coefficient Φ [37]. It switches the particle layer and decomposes and synthesizes the particles to form a global coverage rough particle set $reduct(C_{u_1})$ for the target user. The coverage factor Φ is a value between 0 and 1 that controls the coverage and particle of the user's neighbors. When adjusted specifically, a smaller coverage factor results in a finer user classification, while a larger value makes the classification broader. In order to find the best coverage factor, it can be gradually adjusted according to the feedback of the recommendation effect: if the recommendation is too general, the coverage factor can be appropriately reduced to improve the accuracy; If the recommendation is too limited, the coverage factor can be increased to broaden the scope of the recommendation. In this way, the performance of the recommendation system can be optimized by constantly adjusting the coverage factor. The coverage coefficient is adjusted to obtain the user's coarse and fine grained nearest neighbor k_{u_1} [31]. Based on the above results, it predicts and scores, and the calculation is shown in Equation (17).

$$p_{ui} = \bar{q}_u + \frac{\sum_{v\in k_u} sim(u,v)\times(q_{vi}-\bar{q}_v)}{\sum_{v\in k_u} \left|sim(u,v)\right|} \tag{17}$$

In Equation (17), k_u indicates all the nearest neighbors of the user at the current granularity.

Finally, the algorithm selects games with predicted scores greater than the rating threshold and recommends them to the target users. CRGLC algorithm effectively solves the sparsity problem of user data through the comprehensive application of data preprocessing technology, GC theory, particle partitioning and coverage methods and cluster analysis, thus improving the accuracy of recommendation. CRGLC algorithm firstly cleans, transforms and standardizes user data to ensure data quality. Next, it uses GC theory to build overlay rough particles that represent different user preferences for game types and form a tightly connected network through overlay relationships. Further particle and overlay operations help the algorithm capture nuances and interconnections of user interests at different levels. Finally, the processed particles are classified through cluster analysis, so that users with similar interests are grouped into the same group, and then a personalized game recommendation list is generated according to the common characteristics of these groups and the personal preferences of target users. Through the processing process of CRGLC algorithm, not only effectively overcome the challenges brought by data sparsity, but also greatly improve the intelligence and accuracy of the recommendation system, providing users with a game recommendation experience that is more suitable for their actual needs. The pseudocode for CRGLC and K-means integration is shown in Figure 5.

Figure 5

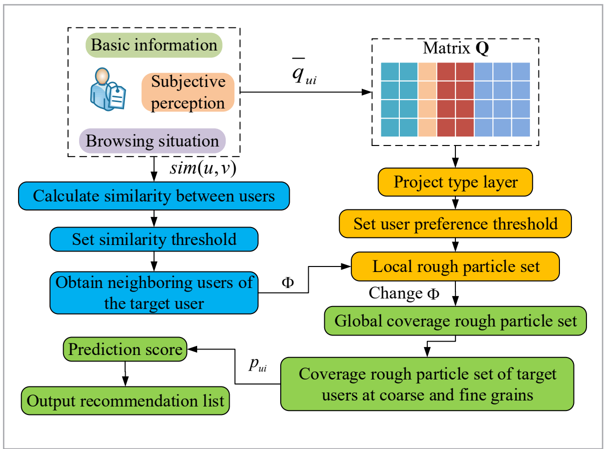
CRGLC and K-means integration pseudocode

```
function CRGLC with K-means clustering

variables:
    data, clusters, centroids

initialize:
    clusters = []
    centroids = random selection of k data points

repeat:
    for each data point in data:
        find the closest centroid and assign the data point to that cl
    for each cluster:
        calculate new centroid based on the points in the cluster
    if centroids are not changed:
        break
    else:
        update centroids

until convergence or max iterations reached

return clusters
```

The difference between this method and the existing particle size calculation method lies in the following three points. First, in the traditional particle size calculation method, the particles are usually obtained directly by data partitioning or clustering. In CRGLC algorithm, the kernel is constructed by introducing the covering rough set theory, which makes the definition and construction of the kernel more flexible and richer. Second, traditional particle size calculation methods tend to focus only on data analysis and processing under a single particle size. The CRGLC algorithm realizes multi-granularity data analysis and processing through granularity division and coverage operation, and can mine information in data more comprehensively. Third, the traditional granularity calculation method mainly focuses on data analysis and mining, and rarely includes the recommendation application. The CRGLC algorithm is specially designed for the recommendation system, and the personalized game recommendation is realized by combining the recommendation algorithm such as collaborative filtering. The recommendation of the CFR model integrating CRGLC is shown in Figure 6.

Figure 6

CFR Process of Integrating CRGLC



KC-CF algorithm is optimized based on K-means CFA recommendation model by introducing particle size calculation theory. The algorithm solves the sparsity problem of user data by constructing the covering rough particle set and mining user preferences comprehensively. At the underlying mathematical level, KC-CF algorithm uses the coverage rough set theory to control the coverage degree and granularity of the user's neighbors by setting the coverage coefficient, so as to realize multi-granularity data analysis and processing. In addition, the algorithm also combines collaborative filtering and other recommendation technologies, constructs a preference matrix according to the user's preference degree, finds the nearest neighbor by calculating the similarity between users, and finally realizes the accurate personalized game recommendation. This optimization method not only improves the accuracy of recommendation, but also makes the recommendation system more intelligent, and can provide users with more suitable game recommendations for their actual needs.

4. Performance Evaluation of Personalized EGR Based on CRGLC Optimization

The research determined the coverage coefficient and the best recommended number of models through mean absolute error (MAE) and Gini index. Then, the classification performance of CRGLC was evaluated using accuracy, precision, recall, and F1 score as indicators. Finally, the clustering and recommendation effects of the model were tested.

4.1. Parameter Determination of CRGLC Algorithm

The recommendation model using K-means CFA was recorded as K-CF, and the recommendation model optimized through GC was recorded as KC-CF. The PCA method was utilized to calculate the weights of each user label. The corresponding eigenvalues, cumulative contribution rate, comprehensive rating coefficient, and indicator weights are expressed in Table 2. The data in Table 2 provides an understanding of user label weights, eigenvalues, cumulative contribution rates, comprehensive score coefficients, and indicator weights. These data were key indicators for evaluating and comparing preferences and characteristics between different users. By analyzing and calculating this data, users' needs and preferences could be better understood and personalized recommendations could be made based on this information. This data could also be used to evaluate and optimize the performance of recommendation algorithms, thereby improving the accuracy of recommendations and user satisfaction.

In order to further explore the optimal value of the coverage factor, the research deliberately selected the Steam game dataset for experimental analysis. The data set was carefully sparsely sampled to ensure the accuracy and reliability of the experimental results, and the data was scientifically divided into the

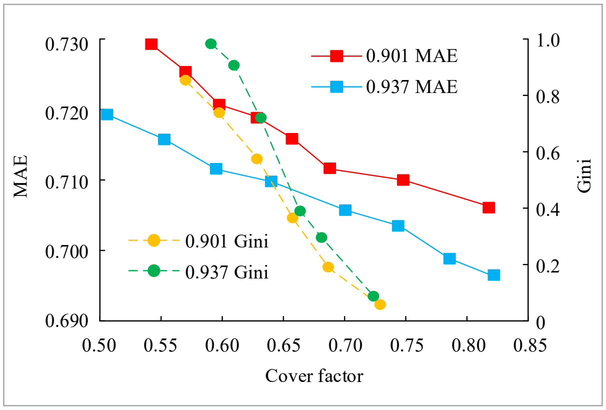
Table 2
User Tag Weights Calculated Using PCA Method

User tag	Eigenvalue	Contribution rate	Comprehensive scoring coefficient	Indicator weight
Age	1.1291	0.0165	0.0138	0.0556
Gender	0.7747	0.0113	0.0061	0.0244
Education	0.4303	0.0063	0.0026	0.0106
Number of views	3.6955	0.0538	0.0414	0.1677
Follow duration	1.9065	0.0278	0.0361	0.1458
Game preferences	55.2330	0.8046	0.0959	0.3881
Satisfaction evaluation	5.4730	0.0797	0.0513	0.2078

training set and the test set according to the ratio of 8:2. The Steam game dataset is derived from Steam, a well-known digital distribution platform that not only offers a wealth of game resources, but also incorporates a variety of social features, making it an ideal choice for studying game recommendation algorithms. The data set used in the research recorded the multiple interactions between users and games in detail, covering a large user group and rich game resources. The data set contains information on tens of thousands of game users from diverse backgrounds and is broadly representative. The number and types of games in the data set are also very large, ranging from classic masterpieces to independent sketches, with diverse types to meet the individual needs of different users. These games cover action, adventure, role playing, strategy, simulation, sports, shooting and other game types, providing a solid foundation for the diversified testing of recommendation algorithms. The sparsity of the Steam game dataset in this study was 0.937 and 0.901, respectively. To verify the optimization effect of KC-CF model, MAE was used as the index to describe the accuracy of the recommended model, and Gini index was used to describe the diversity of the model [8, 22]. The similarity threshold between users was set to 0.04, and the coverage coefficient of the KC-CF model was adjusted on the dataset to analyze the impact of different coverage coefficients on model performance. The results are shown in Figure 7.

In Figure 7, the MAE and Gini values of the model decreased with the increase of coverage coefficient Φ. The smaller the MAE, the higher the accuracy of the model, denoting that changes in coverage coefficient

Figure 7
Experimental Results of the Model on Two Datasets with Different Coverage Coefficients

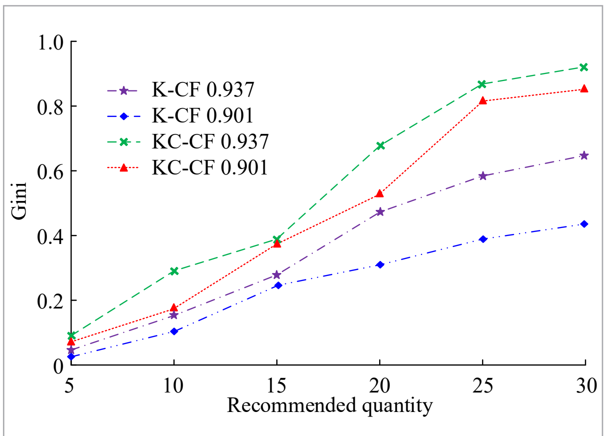


had a significant impact on the accuracy of model recommendations. Compared to the MAE and Gini values in the two graphs, under the same coverage coefficient, the model's recommendation accuracy for datasets with sparsity of 0.901 was close to that of datasets with sparsity of 0.937, indicating that the model could weaken the impact caused by data sparsity. Due to the higher Gini value, the more diverse the recommended games were. To balance the accuracy and diversity of the model, the coverage coefficient at a Gini value of 0.4 was the best coverage coefficient. Therefore, the best coverage coefficient for the Steam game dataset with a sparsity of 0.901 and 0.937 was 0.655 and 0.667, respectively. The corresponding MAE values were 0.716 and 0.708, respectively. According to the above results, when the Gini index was 0.4, the model could better balance accuracy and diversity. Therefore, the optimal coverage coefficients for Steam game datasets with a sparsity of 0.901 and 0.937 were determined to be 0.655 and 0.667, respectively. The reason for the above results was that the coverage factor affects the performance and accuracy of dynamic difficulty adjustment (DDA) system by controlling the granularity of clustering. Smaller coverage coefficients led to finer clustering results, which may improve the performance of DDA systems when dealing with details and local features. However, too small coverage factor may also lead to over-fitting phenomenon and reduce the generalization ability of the system. Conversely, larger coverage coefficients produce broader clustering results, which may improve the performance of DDA systems when dealing with global features and overall structures. However, too large a coverage factor may also lead to under-fitting, making the system unable to fully capture the details of the data. Therefore, choosing the right coverage factor is crucial to the performance and accuracy of a balanced DDA system. In this experiment, the optimal recommendation number of the two models under different data sets was also studied, and the results obtained with Gini value as the evaluation index were shown in Figure 8.

In Figure 8, the Gini coefficients of both algorithms increased with the rising of the amount of recommendations in different datasets. The K-CF algorithm showed the worst recommendation diversity in the 0.901 dataset, while the optimized KC-CF algorithm performed better in the 0.901 dataset than the

Figure 8

Changes in Gini Coefficient under Different Recommended Numbers



K-CF algorithm in both datasets, indicating that the optimized algorithm could effectively alleviate the influence of data sparsity on model recommendation diversity. Considering the time consumption of algorithm operations, the optimal number of recommendations was set to 25, with a corresponding Gini value of 0.820. According to the above research results, the coverage coefficient selected in this study was 0.901.

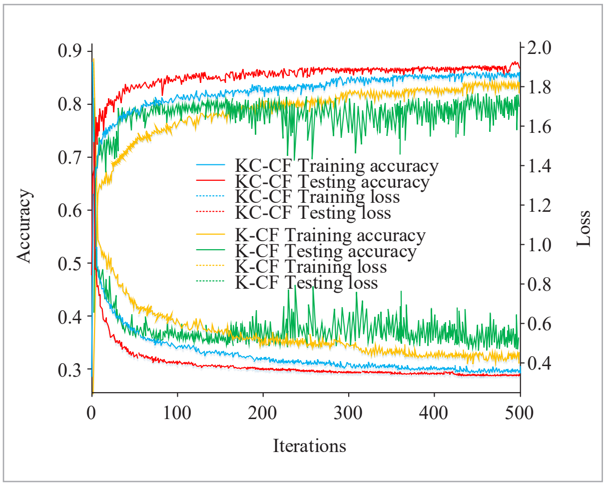
4.2. Analysis of the Classification Effect of Recommendation Models Based on CRGLC

When recommending games to users, the accuracy of the model's classification of games can also affect the accuracy of recommending games [9]. This study divided educational games into role, structure, performance, sports and intellectual games, and music games based on their educational functions. The goal of this experiment was to categorize educational games and recommend suitable games to users. In this context, the classification performance index could directly evaluate the performance of the model in a given category, so as to reflect the performance of the model in practical application. Moreover, clustering performance indicators were usually used in unsupervised learning scenarios, where the real category of data was unknown, and the inherent structure of data needed to be discovered by the algorithm itself. However, in this experiment, there were clear game categories as supervised information, so it is not appropriate to use clustering performance indicators. Therefore, considering the experimental objectives, data characteristics and interpretability of indicators, this study chose to use classification performance indicators (accuracy, loss function, etc.) to evaluate the performance of the model. To verify that the optimized algorithm can complete the game classification under the condition of sparse data, Steam game data set with a sparsity of 0.901 was applied to train and test the model and compare it with the K-CF algorithm. The accuracy and Loss function values are shown in Figure 9.

From Figure 9, the accuracy and Loss function of K-CF algorithm fluctuated greatly. The stability of the algorithm was not strong. the rate of convergence was slow and the effect was poor. Moreover, it was easy to fall into local optimization. The accuracy of its diagnosis was 0.785, which was lower than the training value and had a larger error. This indicated that

Figure 9

Accuracy and Loss Function Values of K-CF and KC-CF Algorithms



its effectiveness in classifying educational games was not satisfactory. KC-CF algorithm had the best convergence effect and the fastest rate of convergence. The accuracy of the test and the training sets obtained by the combined algorithm and the coincidence of the Loss function values were high. The accuracy of the test set reached 0.880, which was higher than the K-CF algorithm and has less error. In order to further verify the classification advantages of the optimization algorithm in the case of sparse data, under the same data set, the accuracy, precision, ROC curve and other indexes of the algorithm were compared with social spatio-temporal probabilistic matrix factorization (SSTPMF), Slope One, and Graph Neural Network-based Recommendation Algorithm (GNN-RA) is compared. The comparison results of accuracy and precision of the four methods are shown in Figure 10.

From Figure 10(a), the accuracy of KC-CF algorithm is 0.880, while the precision of SSPPF algorithm, Slope One algorithm and GNN-RA algorithm were 0.865, 0.843, and 0.796, respectively. In contrast, KC-CF algorithm had higher precision. From Figure 10 (b), the accuracy rates of KC-CF, SSTPMF, Slope One and GNN-RA algorithms were 0.893, 0.852, 0.804 and 0.785, respectively. The above results show that KC-CF algorithm performs better than the comparison algorithm in terms of precision and accuracy. The comparison results of F1 score and ROC curves of the four algorithms are shown in Figure 11.

From Figure 11(a), F1 score of KC-CF model was the highest, which is 0.826, which is better than 0.826 of SSPPF algorithm, 0.804 of Slope One algorithm and 0.779 of GNN-RA. From Figure 11(b), KC-CF model had the highest area under the ROC curve, and the area under the ROC curve of KC-CF algorithm, SSP-PF algorithm, Slope One algorithm and GNN-RA were 0.895, 0.816, 0.774, and 0.718, respectively. The above results show that the KC-CF algorithm proposed in this study has better performance from the two dimensions of F1 score and ROC curve. To visually display the classification status of the model, 1000

Figure 10

Comparison between the results of precision and accuracy changes of the four algorithms

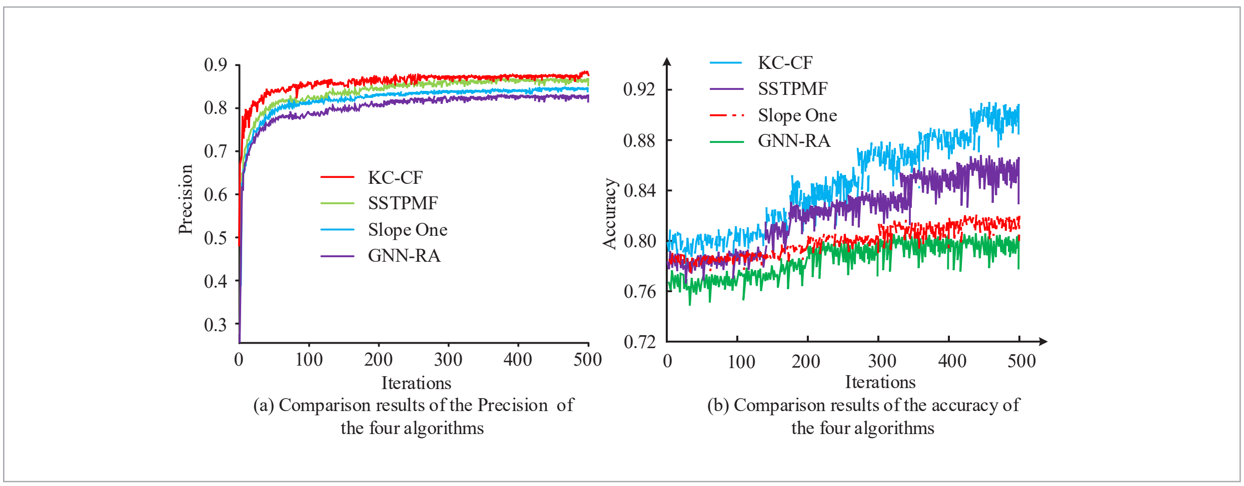


Figure 11

Comparison between the results of F1 score distribution and ROC curve of the four algorithms

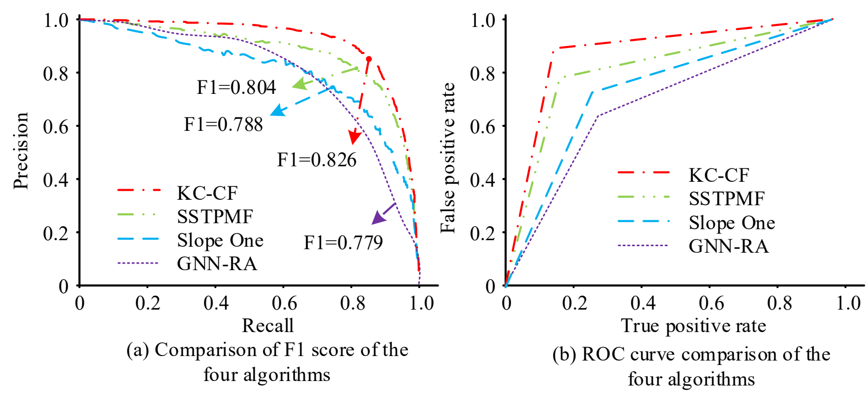
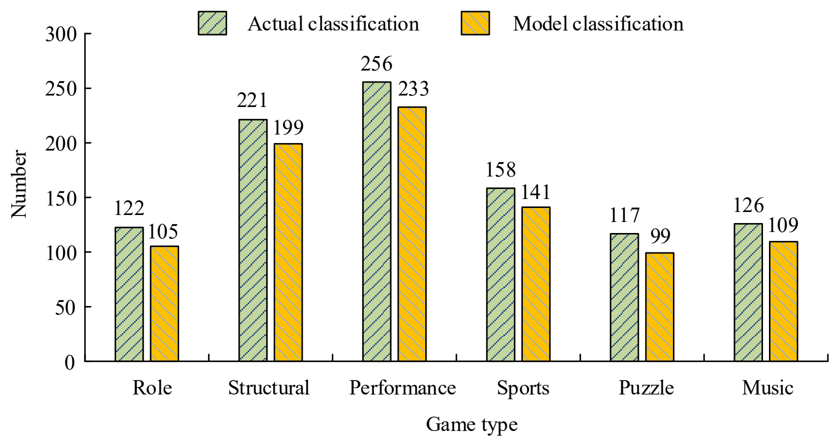


Figure 12

Distribution of the Model's Classification Quantity for 1000 Game Samples



samples were selected from the Steam game dataset with a sparsity of 0.901, and their actual classification status and model classification status were statistically analyzed. The results are shown in Figure 12.

From Figure 12, the model correctly classified 887 out of 1000 educational game samples, which was close to the actual classification results. From the distribution of different types of game numbers, the change in game numbers had little impact on the classification performance of the model, with classification accuracy above 85%. For intelligence games with relatively little data, it still showed high classification accuracy.

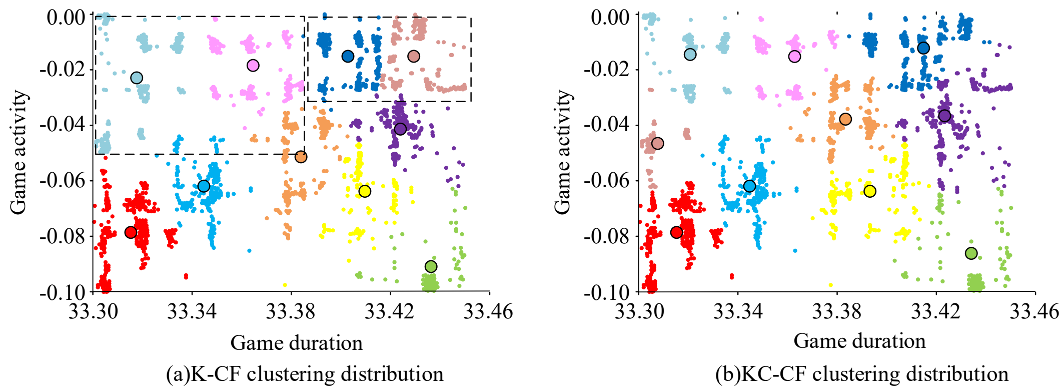
4.3. Application Evaluation of Personalized EGR

To test the actual application performance recommended by the model, based on the above parameters and experimental results, 10 users were selected from Steam game players who were not used for model training, and their neighbors were clustered using KC-CF and K-CF algorithms. The results are shown in Figure 13.

Comparing Figures 13(a)-(b), the coverage range of each cluster center obtained by using the KC-CF was relatively consistent, and they were all located in the

Figure 13

Clustering Results of Two Algorithms for 10 Users

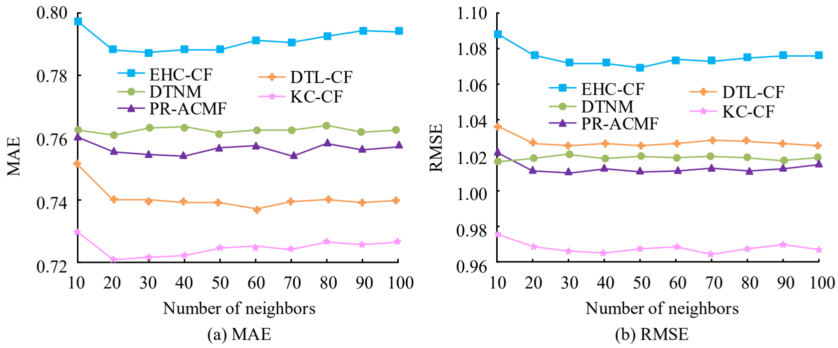


more concentrated positions of this type. The coverage range of K-CF algorithm's clustering centers varied, especially the clustering centers in the dashed box, which were far from the points with dense neighbors, indicating poor clustering performance and scattered neighboring data. By calculating the intra and inter cluster variations of the two algorithms, the intra cluster variation and out of cluster variation of the KC-CF were 0.004537 and 6.360573, while the inter cluster variation and out of cluster variation of the K-CF algorithm were 0.006147 and 6.184695, respectively. By comparison, the KC-CF had smaller variation within the cluster and greater variation outside the cluster, indicating that it could make the data points of similar clusters closer and the differences between different clusters were greater. This proved that the KC-CF algorithm clustering results were more reasonable and effective. In addition, the performance of KC-CF algorithm was compared with PR-ACMF, DTL-CF, DTNM and EHC-CF. To ensure the fairness of the experiment and the reliability of the results, all algorithms were evaluated in a unified experimental environment using the same training and test data sets. The implicit feature dimension of PR-ACMF algorithm was set to 100 and the learning rate was 0.01. DTL-CF algorithm adopted three-layer neural network structure, and the initial learning rate was 0.001. DTNM algorithm combined deep learning and matrix decomposition, and its matrix decomposition dimension was 50. The embedding dimension of EHC-CF algorithm was 64 and the learning rate was 0.005. The KC-CF algorithm was optimized by initializing parameters, using grid search method to find the best parameter combination, testing parameter combination, etc. The nearest neighbor number was determined to be 20, the learning rate was 0.01, and the regularization coefficient was 0.001. The comparison indexes were MAE, RMSE and complexity. The study conducted several experiments on a dataset of Steam games not used for model training, averaging the final results. The performance changes of algorithms with different nearest neighbors are shown in Figure 14.

From Figure 14, the MAE and RMSE values did not change much with the increase of the amount of nearest neighbors and remained fluctuating within a certain range. Especially after the amount of nearest neighbors reached 20, the changes in MAE and RMSE values generally tended to flatten out. The MAE value of the KC-CF algorithm mainly fluctuated between 0.72-0.73, while the RMSE value fluctuated between 0.96-0.98. Compared with the other four algorithms, the KC-CF algorithm had the lowest MAE and RMSE values at any number of nearest neighbors. This indicated that the prediction score of the algorithm had superiority. The above experimental results show that KC-CF algorithm has the lowest MAE and RMSE values for any number of near neighbors, which indicates that its prediction performance is better than the other four algorithms. This advantage may be due to the KC-CF algorithm's ability to reasonably determine the clustering center and thus obtain better clustering results, which helps to more accurately predict

Figure 14

MAE and RMSE Values of Different Algorithms under Different Number of Nearest Neighbors



the score of the target user based on the neighboring user. In addition, from the point of view of algorithm design, KC-CF algorithm may achieve lower computational complexity by optimizing the clustering process. Due to KC-CF's optimization in clustering, it is more efficient in processing large-scale data sets. This efficiency is not only reflected in prediction accuracy, but also in computational speed and resource consumption. In summary, KC-CF algorithm shows advantages in both prediction performance and computational complexity. It not only improves the prediction accuracy, but also reduces the computational complexity, making the algorithm more efficient and accurate when dealing with the task of recommendation system through reasonable clustering center determination and optimization. Therefore, the performance of KC-CF algorithm is significantly better than other comparison methods.

To visualize how well the model predicts user ratings, it took a performance game as an example, and selected 500 users who have rated that type of game from a Steam game dataset that was not used for model training. Models were used to predict user ratings and the results were compared with actual ratings. User ratings were scored out of 10, with higher scores indicating higher user satisfaction with the game. First, the study divided consecutive scores into five discrete scores of 0-2, 2-4, 4-6, 6-8, and 8-10. The distribution of actual and predicted scores was then calculated, and a 5x5 confusion matrix was constructed, where the rows represent the actual score segments and the columns represent the predicted score segments. The confusion matrix of actual and predicted ratings is shown in Table 3.

Through the analysis of the confusion matrix in Table 3, the model showed certain accuracy in predicting user ratings. Among them, the values on the diagonal were all higher, which were 0.987, 0.991, 0.983, 0.975 and 0.977, respectively. This result shows that the model can correctly predict the user's score in most scores. Finally, to fully evaluate the performance of the proposed DDA system, the study compared it with

Table 3

Confusion matrix of actual and predicted scores

/	Prediction 0-2	Prediction 2-4	Prediction 4-6	Prediction 6-8	Prediction 8-10
Actually 0-2	0.987	0.258	0.312	0.287	0.117
Actually 2-4	0.325	0.991	0.226	0.213	0.058
Actually 4-6	0.531	0.221	0.983	0.268	0.147
Actually 6-8	0.218	0.189	0.156	0.975	0.311
Actually 8-10	0.311	0.115	0.147	0.189	0.977

Table 4
Performance Comparison of the Different DDA Systems

System name	Use technology	Prediction accuracy	Recommendation diversity	Run time (s)	Memory consumption (MB)
Proposed DDA	GC optimization + collaborative filtering	0.85	0.70	120	800
DDA-RL [29]	Reinforcement learning	0.80	0.65	150	900
DDA-EA [2]	Evolutionary algorithm	0.82	0.68	180	1000
DDA-NN [12]	Neural network	0.83	0.66	100	1200
DDA-Baseline [23]	Traditional collaborative filtering	0.78	0.60	90	700

four other state-of-the-art DDA systems using different technologies. Four key performance metrics were selected: prediction accuracy, recommendation diversity, runtime, and memory consumption. Table 4 shows the specific comparison results.

Table 4 shows the results of five different DDA systems on four key performance indicators. Among them, Proposed DDA represents the proposed system based on GC optimization and collaborative filtration. From the data in Table 4, the prediction accuracy rate of Proposed DDA was 0.85, which was higher than the other four systems, indicating its superior prediction ability. In terms of recommendation diversity, Proposed DDA also led with a score of 0.70, indicating that its recommendation results were more diverse. Although the Proposed DDA run time of 120 seconds was not the shortest, it was still within a reasonable range compared to similar performance systems. Proposed DDA memory consumption of 800MB was moderate, indicating that it was relatively efficient in resource utilization. To sum up, Proposed DDA had outstanding performance in prediction accuracy and recommendation diversity, and maintained a good balance in running time and memory consumption, which verified its effectiveness as an advanced DDA system.

4.4. Discussion

According to the above results, the reasons for the superiority of the proposed method were analyzed. Firstly, KC-CF algorithm successfully weakened the effect of data sparsity and improved the performance of the model on sparse data set by introducing granularity calculation and covering rough set theory. This optimization strategy allows the model to capture user interests more accurately and generate effective recommendations even when user behavior data is limited. Secondly, KC-CF algorithm ensures the stability and accuracy of the model under different nearest neighbors through reasonable parameter optimization and experimental design. In particular, the optimal parameter combination determined by grid search method enables the model to maintain the prediction accuracy, reduce the computational complexity, and improve the practicality and scalability of the model. This result was similar to the study conducted by Li and Zhai et al. [13].

From the perspective of the practical application of game recommendations, the KC-CF algorithm's high prediction accuracy and recommendation diversity are crucial to improving user experience and engagement. This result coincides with the research results of Li and Liu's team [14]. Accurate predictions mean that the system is able to provide users with game recommendations that are more in line with their personal preferences, which increases user satisfaction. The diversity of recommendations helps prevent users from falling into information cocoons, increasing the opportunity for users to explore new games, and further increasing user engagement. In addition, the impact of KC-CF algorithm improvements on actual user experience and recommendation adoption cannot be ignored. By providing more accurate and diverse recommendations, the system can better meet the individual needs of users and enhance users' trust and dependence on the recommendation system. This will not only help improve user satisfaction, but may also encourage users to use the recommendation system more frequently, further increasing user engagement. Finally, in order to improve user satisfaction and engagement, the research suggests continuous optimization of KC-CF algorithm, exploration of more effective data sparsity processing methods, and consideration of incorporating more user behavior data into model training. In addition, real-time adjustment combined with user feedback is also an important way to improve model performance. Through the above measures, it is expected to provide users with more intelligent and personalized game recommendation services, so as to achieve greater success in the field of game recommendation.

5. Conclusion

To solve the problem of sparse data in EGR, an optimized CFR algorithm based on GC and K-means clustering was proposed. PCA method was utilized to calculate user tag weights and build CRGS. Experiment results showed that the algorithm achieved good prediction performance on datasets with sparsity of 0.937 and 0.901, MAE values of 0.708 and 0.716, respectively. It also showed high accuracy, precision and F1 score in game classification. In addition, the algorithm had advantages over other algorithms in recommending educational game types. However, the study also revealed the limitations and challenges of the algorithm, including the handling of noisy and lost data, cold start issues, and the privacy and security of user data. In response to these limitations, future research should focus on exploring how machine learning and deep learning techniques can be used to improve the robustness and accuracy of algorithms, especially in dealing with noise and lost data. At the same time, the research should further explore diverse data sources and characteristics to solve the cold start problem and ensure the accuracy and personalization of the recommendation system. In addition, the privacy and security protection of user data is also an indispensable part of future research, and it is necessary to develop effective privacy protection and data security policies.

Funding

The research is supported by: Planning topic of Shaanxi Academy of Teaching Sciences: Research on the Reform and Influencing Factors of Informatization Teaching Methods of University Teachers in the Post-epidemic Era (SGH22Y1778).

Conflict of Interest

The authors declare that they have no conflict of interest.

Data Availability Statement

The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

1. Alhaj, M. M. A., Shambour, Q. Y., Tahrawi, M. M. A. A Hybrid Collaborative Filtering Recommendation Algorithm for Requirements Elicitation. International Journal of Computer Applications in Technology, 2020, 63(1/2), 135-146. https://doi.org/10.1504/IJCAT.2020.10030077
2. Bello, M., Nápoles, G., Vanhoof, K., Bello, R. Data Quality Measures Based on Granular Computing for Multi-Label Classification. Information Sciences, 2021, 560(8), 51-67. https://doi.org/10.1016/j.ins.2021.01.027
3. Chen, J., Zhu, T., Gong, M., Wang, Z. A Game-Based Evolutionary Clustering with Historical Information Aggregation for Personal Recommendation. IEEE Transactions on Emerging Topics in Computational Intelligence, 2023, 7(2), 552-564. https://doi.org/10.1109/TETCI.2022.3189084
4. Davtalab, M., Alesheikh, A. A. A POI Recommendation Approach Integrating Social Spatio-Temporal Information Into Probabilistic Matrix Factorization. Knowledge and Information Systems, 2021, 63(1), 65-85. https://doi.org/10.1007/s10115-020-01509-5
5. Deboeuf, S., Fall, A. Cohesion and Aggregates in Unsaturated Wet Granular Flows Down a Rough Incline. Journal of Rheology, 2023, 67(4), 909-922. https://doi.org/10.1122/8.0000631
6. Demirkiran, E. T., Pak, M. Y., Çekik, R. Multi-Criteria Collaborative Filtering Using Rough Sets Theory. Journal of Intelligent and Fuzzy Systems, 2021, 40(1), 907-917. https://doi.org/10.3233/JIFS-201073
7. Ejegwa, P. A., Agbetayo, J. M. Similarity-Distance Decision-Making Technique and Its Applications via Intuitionistic Fuzzy Pairs. Journal of Computational and Cognitive Engineering, 2023, 2(1), 68-74. https://doi.org/10.47852/bonviewJCCE512522514
8. Gasmi, I., Azizi, M. W., Seridi-Bouchelaghem, H., Azizi, N., Belhaouari, S. B. Enhanced Context-Aware Recom-

mendation Using Topic Modeling and Particle Swarm Optimization. Journal of Intelligent and Fuzzy Systems, 2021, 60(4), 12227-12242. https://doi.org/10.3233/JIFS-210331

9. Gharibi, R., Safdel, A., Fakhrahmad, S. M., Sadreddini, M. H. A Content-Based Model for Tag Recommendation in Software Information Sites. The Computer Journal, 2019, 64(11), 1680-1691. https://doi.org/10.1093/comjnl/bxz144
10. Huang, J., Ma, B., Wang, M., Zhou, X. K., Yao, L., Wang, S. J., Qi, L. Y., Chen, Y. Incentive Mechanism Design of Federated Learning for Recommendation Systems in MEC. IEEE Transactions on Consumer Electronics, 2024, 70(1), 2596-2607. https://doi.org/10.1109/TCE.2023.3342187
11. Jain, P., Som, T. Multigranular Rough Set Model Based on Robust Intuitionistic Fuzzy Covering with Application to Feature Selection. International Journal of Approximate Reasoning, 2023, 156(5), 16-37. https://doi.org/10.1016/j.ijar.2023.02.004
12. Lee, H., Choi, Y., Eizad, A., Song, W.-K., Kim, K.-J., Yoon, J. A Machine Learning-Based Initial Difficulty Level Adjustment Method for Balance Exercise on a Trunk Rehabilitation Robot. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 2023, 31(5), 1857-1866. https://doi.org/10.1109/TNSRE.2023.3260815
13. Li, W., Zhai, S., Xu, W., Pedrycz, W., Qian, Y., Ding, W., Zhan, T. Feature Selection Approach Based on Improved Fuzzy C-Means with Principle of Refined Justifiable Granularity. IEEE Transactions on Fuzzy Systems, 2023, 31(7), 2112-2126. https://doi.org/10.1109/TFUZZ.2022.3217377
14. Li, Y., Liu, K., Satapathy, R., Wang, S., Cambria, E. Recent Developments in Recommender Systems: A Survey. IEEE Computational Intelligence Magazine, 2024, 19(2), 78-95. https://doi.org/10.1109/MCI.2024.3363984
15. Liang, H., Cai, R. A New Correlation Coefficient of BPA Based on Generalized Information Quality. International Journal of Intelligent Systems, 2021, 36(9), 4775-4794. https://doi.org/10.1002/int.22490
16. Liang, M., Mi, J., Feng, T. Optimal Granulation Selection for Similarity Measure-Based Multigranulation Intuitionistic Fuzzy Decision-Theoretic Rough Sets. Journal of Intelligent and Fuzzy Systems, 2019, 36(3), 2495-2509. https://doi.org/10.3233/JIFS-181193
17. Liu, C., Cai, K., Miao, D., Qian, J. Novel Matrix-Based Approaches to Computing Minimal and Maximal Descriptions in Covering-Based Rough Sets. Information Sciences, 2020, 539(9/10), 312-326. https://doi.org/10.1016/j.ins.2020.06.022
18. Liu, M., Zhang, B., Li, X., Tang, W., Zhang, G. Q. An Optimized K-Means Algorithm Based on Information Entropy. The Computer Journal, 2021, 64(7), 1130-1143. https://doi.org/10.1093/comjnl/bxab078
19. Liu, N., Zhao, J. Recommendation System Based on Deep Sentiment Analysis and Matrix Factorization. IEEE Access, 2023, 11, 16994-17001. https://doi.org/10.1109/ACCESS.2023.3246060
20. Liu, T., Deng, X., He, Z., Long, Y. TCD-CF: Triple Cross-Domain Collaborative Filtering Recommendation. Pattern Recognition Letters, 2021, 149(6), 185-192. https://doi.org/10.1016/j.patrec.2021.06.016
21. Liu, X., Sun, J., Zhang, Q., Wang, Z., Xu, Z. Learning to Learn Evolutionary Algorithm: A Learnable Differential Evolution. IEEE Transactions on Emerging Topics in Computational Intelligence, 2023, 7(6), 1605-1620. https://doi.org/10.1109/TETCI.2023.3251441
22. Matt, C., Hess, T., Weiß, C. A Factual and Perceptional Framework for Assessing Diversity Effects of Online Recommender Systems. Internet Research, 2019, 29(6), 1526-1550. https://doi.org/10.1108/INTR-06-2018-0274
23. Moschovitis, P., Denisova, A. Keep Calm and Aim for the Head: Biofeedback-Controlled Dynamic Difficulty Adjustment in a Horror Game. IEEE Transactions on Games, 2023, 15(3), 368-377. https://doi.org/10.1109/TG.2022.3179842
24. Shafiq, M., Yadav, R., Javed, A. R., Mohsan, S. A. H. CoopGBFS: A Federated Learning and Game-Theoretic-Based Approach for Personalized Security, Recommendation in 5G Beyond IoT Environments for Consumer Electronics. IEEE Transactions on Consumer Electronics, 2024, 70(1), 2648-2656. https://doi.org/10.1109/TCE.2023.3305508
25. Song, Y. T., Wu, S. Slope One Recommendation Algorithm Based on User Clustering and Scoring Preferences. Procedia Computer Science, 2020, 166(1), 539-545. https://doi.org/10.1016/j.procs.2020.02.042
26. Thongchotchat, V., Kudo, Y., Okada, Y., Sato, K. Educational Recommendation System Utilizing Learning Styles, A Systematic Literature Review. IEEE Access, 2023, 11, 8988-8999. https://doi.org/10.1109/ACCESS.2023.3238417
27. Tuah, N. M., Goh, D. L., Nasirin, S., Ahmedy, F., Hossin, M. Mapping Data Mining Technique and Gamification Approach for Studying Post-Stroke Rehabilitation Training, A Systematic Literature Review. IEEE Ac-

cess, 2023, 11(1), 31323-31340. https://doi.org/10.1109/ACCESS.2023.3262260

28. Ünver, M., Olgun, M., Türkarslan, E. Cosine and Cotangent Similarity Measures Based on Choquet Integral for Spherical Fuzzy Sets and Applications to Pattern Recognition. Journal of Computational and Cognitive Engineering, 2022, 1(1), 21-31. https://doi.org/10.47852/bonviewJCCE2022010105

29. Wang, D., Gao, N., Liu, D., Li, J., Lewis, F. L. Recent Progress in Reinforcement Learning and Adaptive Dynamic Programming for Advanced Control Applications. IEEE/CAA Journal of Automatica Sinica, 2024, 11(1), 18-36. https://doi.org/10.1109/JAS.2023.123843

30. Wang, Z., Yue, H., Deng, J. An Uncertainty Measure Based on Lower and Upper Approximations for Generalized Rough Set Models. Fundamenta Informaticae, 2019, 166(3), 273-296. https://doi.org/10.3233/FI-2019-1803

31. Wu, Z., Wang, H., Chen, N., Luo, J. Semi-Monolayer Covering Rough Set on Set-Valued Information Systems and Its Efficient Computation. International Journal of Approximate Reasoning, 2021, 130(2), 83-106. https://doi.org/10.1016/j.ijar.2020.12.011

32. Yang, B., Hu, B. Q., Qiao, J. Three-Way Decisions with Rough Membership Functions in Covering Approximation Space. Fundamenta Informaticae, 2019, 165(2), 157-191. https://doi.org/10.3233/FI-2019-1780

33. Yao, G., Wu, Y., Huang, X., Ma, Q., Du, J. Clustering of Typical Wind Power Scenarios Based on K-Means Clustering Algorithm and Improved Artificial Bee Colony Algorithm. IEEE Access, 2022, 10(6), 98752-98760. https://doi.org/10.1109/ACCESS.2022.3203695

34. Yu, Z., Gao, M., Wang, L. The Effect of Educational Games on Learning Outcomes, Student Motivation, Engagement, and Satisfaction. Journal of Educational Computing Research, 2021, 59(3), 522-546. https://doi.org/10.1177/0735633120969214

35. Zeeshan, Z., Ain, Q. U., Bhatti, U. A., Memon, W. H., Ali, S., Nawaz, S. A., Nizamani, M. M., Mehmood, A., Bhatti, M. A., Shoukat, M. U. Feature-Based Multi-Criteria Recommendation System Using a Weighted Approach with Ranking Correlation. Intelligent Data Analysis, 2021, 25(4), 1013-1029. https://doi.org/10.3233/IDA-205388

36. Zhang, E., Li, H., Huang, Y., Hong, S., Zhao, L., Ji, C. Practical Multi-Party Private Collaborative k-Means Clustering. Neurocomputing, 2022, 467(7), 256-265. https://doi.org/10.1016/j.neucom.2021.09.050

37. Zhang, Y. L., Li, C. Q., Li, J. On Characterizations of a Pair of Covering-Based Approximation Operators. Soft Computing, 2019, 23(12), 3965-3972. https://doi.org/10.1007/s00500-018-3321-8

38. Zhao, Z. Maps Between Covering Approximation Spaces and the Product Space of Two Covering Approximation Spaces. International Journal of Approximate Reasoning, 2019, 115(12), 144-156. https://doi.org/10.1016/j.ijar.2019.09.008

39. Zhou, J. J., Li, X. Y. Hesitant Fuzzy β Covering Rough Sets and Applications in Multi-Attribute Decision Making. Journal of Intelligent Fuzzy Systems, 2021, 41(1), 2387-2402. https://doi.org/10.3233/JIFS-190959