Intelligent Archive Search System Using Cuckoo Search-Enhanced K-Prototypes Clustering

Jianhua Fu
Archives, Beihua University, Jilin, 132013, China
E-mail: fjh102834@163.com

Keywords: K-prototypes algorithm, intelligent search system, cuckoo search, archives, data clustering

Recieved: February 20, 2025

With the rapid advancement of the times, the amount of data stored in archives is increasing. Aiming at the problem of low data search accuracy of traditional search algorithms in archives, this research proposes an intelligent search method for archives that uses cuckoo search algorithm to improve K-prototypes clustering algorithm. First, the Cuckoo search algorithm is innovatively utilized to improve the K-prototype clustering algorithm and solve the problem of poor searching ability of the K prototype algorithm. Then, the CS-K-prototypes algorithm is introduced for intelligent data search in archives. Finally, experiments are conducted using 200 sets of data collected from machine learning repositories as well as real data from a large archive, and performance evaluation metrics such as Precision-Recall curve, MAE, RMSE, Jacquard Coefficient (JC), Rand Index (RI), and Fowlkes Mallows Scores are used. The experiment findings denote that the K-prototypes clustering algorithm proposed by the research combined with the cuckoo search algorithm has an accuracy and recall curve offline area of 0.9744 when conducting intelligent search of archive data. In contrast with the K-value average clustering algorithm, the accuracy and recall curve offline area increase by 0.073. Compared to the artificial bee colony algorithm, it improves by 0.2252 under conditions of abundant data. In the practical application experiment of the algorithm model, the proposed model achieves a search accuracy of 97.67%. The above results indicate that the improved K-prototypes clustering algorithm proposed by the research can improve the search accuracy of archive data.

Povzetek: Študija predlaga inteligenten iskalni sistem za arhive z algoritmom CS-K-prototipi, ki izboljša združevanje K-prototipi z algoritmom kukavičje iskanje (CS). Algoritem je primeren za mešane (numerične/kategorične) arhivske podatke.

1 Introduction

As the artificial intelligence algorithms rapidly develop, the frequency of using intelligent algorithms for data search in archives is gradually increasing [1]. The data in the archives is complex and diverse, storing personnel files, document files, accounting files, and so on. These diverse types of archival data need to be clustered and classified for storage in the archives, in order to facilitate subsequent retrieval of data within the archives [2]. The biggest challenge currently is how to perform clustering and classification operations on archival data, as well as retrieve data within the archives. The K-prototypes algorithm is a clustering analysis method that combines K-means and K-modes, particularly suitable for handling mixed data such as archive data, including both numerical and categorical data [3-4]. However, due to the complexity of data in archives, traditional clustering algorithms have low search accuracy and poor classification performance in intelligent search of archives, making it difficult to meet the efficient retrieval needs of archive data management. Although the K-prototypes algorithm can handle mixed data clustering and classification, its search ability is insufficient, which limits its application effect in the intelligent search of archives [5]. The Cuckoo Search (CS) algorithm, as an intelligent search algorithm, has strong global search ability, few parameters, and fast convergence speed [6-7]. Therefore, this study assumes that introducing the CS algorithm to improve the K-prototypes algorithm can effectively enhance the search accuracy and efficiency of the algorithm, thereby better meeting the intelligent search needs of complex data in archives, improving the clustering classification effect and retrieval accuracy of archive data.

2 Related work

With the gradual increase of data in archives, how to search for data in archives has received widespread attention from scholars at home and abroad. Zhu et al. stated that clustering algorithms are mainly used in data retrieval to group unlabeled data to extract meaningful information. A review of clustering algorithms was also conducted and it was found that determining the optimal number of clusters will require new feature extraction methods, validation metrics, and clustering techniques as the amount of clustered data increases and changes [8].

Wang et al. proposed a calibration management system based on data mining and clustering algorithms to improve the ability of archives to retrieve information and share resources. The results showed that the proposed method could improve the accuracy of book recommendation, which was 16.7% higher than the traditional method on average [9]. Amin et al. proposed a personalized book recommendation system based on K-means clustering and association rule techniques to improve the service quality of archives and used Apriori algorithm to build a recommendation model for each cluster. The results showed that the proposed method increased the average accuracy by 3.61% and the average recall by 3.61% compared to the recommendation model without clustering method [10].

In addition, the application of clustering algorithms has also received widespread attention from scholars both domestically and internationally. To solve the optimization problem of numerical functions Minh et al. proposed a meta heuristic optimization algorithm based on K-means clustering algorithm. Cluster centroid vectors were established through K-means clustering algorithm, and the movement strategy was determined through data iteration. The findings showed that the accuracy of the proposed K-means clustering meta heuristic optimization algorithm reached 94.32% [11]. Zhang et al. proposed a fuzzy mean clustering algorithm to address the issue of high computational complexity in traditional algorithms. By dividing the input image into multiple filtering windows, pixels were fitted into corresponding neighborhoods to obtain pixel spatial information, and filter windows and generalized neighborhood windows were reduced [12]. Shi et al. proposed a weighted fuzzy K-prototypes algorithm for mixed data by constructing a consumer multi-perspective segmentation index system for achieving consumer segmentation and combining particle swarm optimization algorithm and sparrow search algorithm. The results showed that the proposed method had a good effect on the synthesis of evaluation indexes [13]. Hafid et al. analyzed K-Medoids and K-prototypes methods for grouping patients based on complex clinical data. The results showed that the K-prototypes algorithm could detect early hypertension risk by categorizing patients into different risk groups, but the clustering results were still weak [14].

Table 1: Related work summary table

Literature	Method	Data set	Result	Limitation
[8]	Overview of clustering algorithms	Cross disciplinary datasets	-	Unsolved high-dimensional mixed data processing problem
[9]	Data mining+clustering algorithm	Library borrowing records	The accuracy of book recommendations has increased by 16.7%	Not considering multimedia files such as images and audio
[10]	K-means clustering+Apriori association rules	Library user behavior data	The average accuracy has increased by 3.61%	Cold start issue not resolved
[11]	K-means+Meta heuristic Optimization Algorithm	Structural damage dataset	Recognition accuracy 94.32%	Not sensitive to classification attributes
[12]	Fuzzy clustering	BSDS500 Image Collection	The split Intersection over Union is 0.82	High memory consumption and poor real-time performance
[13]	Weighted fuzzy K-prototypes+PSO/SSA	Consumer data of fresh food market	The contour coefficient is 0.68	Parameter sensitive, only applicable to commercial scenarios
[14]	K-Medoids and K-prototypes	Clinical data of hypertension	The F1 value identified by the risk group is 0.71	Weak clustering results, high sensitivity to small samples

In summary, although previous researchers have done a lot of work on clustering algorithms, most of the studies mainly focus on specific application scenarios and have not fully explored the performance differences of different clustering algorithms in dealing with hybrid archive data, and the specific performance in dealing with archive data still needs to be further verified. Therefore, in this study, K-prototypes algorithm is chosen as the basis, which can effectively deal with hybrid attributes in archive data. Compared with hybrid clustering techniques such as Density-Based Spatial Clustering of Applications with Noise (DBSCAN), the K-prototypes algorithm has higher efficiency and scalability in dealing with large-scale archive data, and can better adapt to the diversity and complexity of archive data. In addition, to further improve the clustering effect, this study introduces the CS algorithm to optimize the K-prototypes algorithm, which solves the randomness and uncertainty problems of the

K-prototypes algorithm in determining the initial clustering centers, to improve the stability and accuracy of clustering. This study not only fills the gap of existing research in dealing with hybrid archival data, but also provides a more efficient and reliable solution for archive data retrieval.

3 The intelligent search design of improved clustering algorithms in archives

3.1 Design of improved clustering algorithm for CS algorithm

In the practical application of clustering algorithms, there are both numerical attribute datasets and categorical attribute datasets, making it difficult to classify different datasets [15-16]. To enhance the intelligent search capability of archives in the archives, the K-prototypes algorithm is introduced in the research, aiming to use this algorithm to process large datasets and high-dimensional data, and improve classification performance. The K-prototypes algorithm is mainly composed of K-Means and K-Modes algorithms, and is mainly used to handle clustering problems of mixed data. This algorithm mainly processes mixed data through the objective function, which is mathematically expressed as equation (1).

$$F(W,Z)=\sum_{j=1}^{k}\sum_{i=1}^{n}\left(w_{ij}\right)\left(\sum_{r=1}^{p}\left(x_{ir}^{c}-z_{ir}^{c}\right)^{2}+\gamma\cdot\sum_{r=p+1}^{m}\delta\left(x_{ir}^{s},z_{ir}^{s}\right)\right) \quad (1)$$

In equation (1), $F(W,Z)$ represents the objective function, w_{ij} represents the elements in the partition matrix, γ represents the proportional size of the weights, $\delta\left(x_{ir}^{s},z_{ir}^{s}\right)$ represents the dissimilarity measure, k represents the number of cluster centers, n represents the number of samples in the dataset, m represents the number of categorical attributes, and p represents the number of numerical attributes. x_{ir}^{c} and z_{jr}^{c} represent the values of sample i and cluster center j on the numerical attribute r, respectively. On the basis of the objective function, the data clustering center is calculated, and its expression is shown in Equation (2).

$$z_1=\underset{y\in CA_x}{Max}\left(Dens(y)+d(y,Z)-d(y,C_1)\right) \quad (2)$$

In equation (2), C_1 represents the Cluster specimens. to analyze the attribute values of the data samples, the algorithm calculates the cluster center and expresses it as shown in equation (3).

$$Z_l=\left\{z_1^{c},z_2^{c},\cdots z_p^{c},z_{p+1}^{s}\right\} \quad (3)$$

In equation (3), Z_l represents the cluster center. Among them, the elements in the membership matrix can be represented by formulas, and the mathematical expression is shown in equation (4).

$$w_{ij}=1,d\left(X_i,Z_j\right)\le d\left(X_i,Z_j\right) \quad (4)$$

In equation (4), X_i and Z_j denote sample objects. The dissimilarity is calculated based on the calculated sample object, as expressed in equation (5).

$$d_l\left(X_i,Y_j\right)=\sum_{r=1}^{p}\left(X_i^{c}-Y_j^{c}\right)^{2} \quad (5)$$

In equation (5), X_i^{c} means the categorical attribute of the i th sample, and Y_j^{c} represents the categorical attribute of the j th sample. When the K-prototypes algorithm processes the mixed data, it takes the input of the data and normalizes the results output with the fuzzy matrix. The K-prototypes algorithm has a good clustering and classification effect on data management in archives, but its search ability is poor. To solve this problem, the CS algorithm is introduced to improve the K-prototypes algorithm. The CS algorithm, as a population-based intelligent optimization algorithm that mimics the hatching behavior of cuckoo birds, can improve the search ability of clustering algorithms [17-18]. By introducing the Levy flights mechanism of the CS algorithm to update positions, the global search capability of the K-prototypes algorithm is enhanced, avoiding getting stuck in local optima and improving the search performance of the clustering algorithm, as shown in equation (6).

$$x_i^{t+1}=x_i^{t}+a\oplus Levy(\beta) \quad (6)$$

In equation (6), x_i^{t} represents the i th solution of the t th generation cuckoo position, and $Levy(\beta)$ represents the random path of the cuckoo during the search. The random search path of the CS algorithm is calculated to ensure that the algorithm can effectively search globally and avoid local optima, as shown in equation (7).

$$Levy(\beta)=\frac{\phi\cdot u}{|v|^{\frac{1}{\beta}}} \quad (7)$$

In equation (7), u and v represent constants and both belong to the standard distribution. When the CS algorithm calculates, it mainly initializes the position, then calculates the adaptive value, and finally updates the position and iterates according to the adaptive value.

After the CS algorithm completes the path search, it will calculate the fitness value of the bird's nest to evaluate the quality of the current solution and guide the search direction of the algorithm, which is expressed as equation (8).

$$x_i^{t+1} = x_t^t + r_1(x_j^t + x_k^t) \quad (8)$$

In equation (8), r_1 represents a constant with a value range of $r_1 \in [0,1]$, while x_k^t and x_j^t represent the solutions obtained by the CS algorithm in the t th generation. The process of optimizing the K-prototypes algorithm using the CS algorithm is denoted in Figure 1.

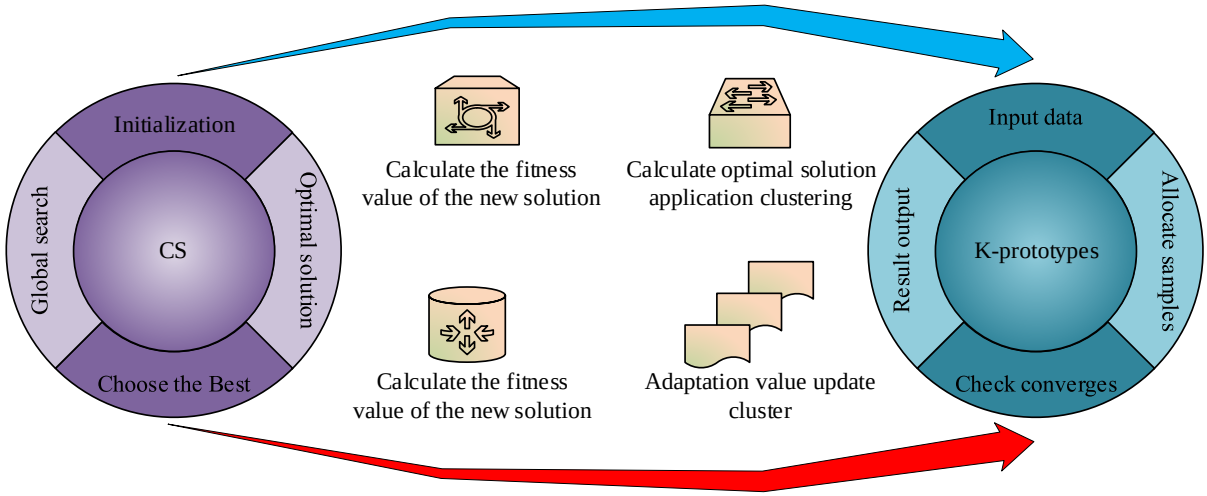


Figure 1: CS optimization K-prototypes process

As shown in Figure 1, the CS-K-prototypes algorithm first receives mixed data and performs data preprocessing. Secondly, the parameters of the CS algorithm is initialized based on the characteristics of the input data. Then, a global search operation is performed. The Lévy flight mechanism is utilized to update the position of the cuckoo's nest and explore the solution space. Subsequently, the optimal solution is selected. The fitness value of each new scheme is calculated and the current optimal scheme is selected based on the fitness value. Next, the clustering center is updated. The optimal solution is applied to update the clustering centers of the K-prototypes algorithm, and the updated clustering centers are applied to mixed data for sample allocation. Finally, the convergence condition is checked and the final clustering results are output. Complexity analysis is conducted on the algorithm. It assumes that the number of iterations is I, the number of samples is N, the feature dimension is D, and the number of cluster centers is K, the total time complexity of the K-means algorithm is $O(I \times N \times K \times D)$. The K-prototypes algorithm is a combination of K-means and K-modes algorithms used for processing mixed data. It assumes that the dimension of numerical attributes is D_n, the dimension of categorical attributes is D_c, and the total time complexity of the K-prototypes algorithm is $O[I \times N \times K \times (D_n + D_c)]$. The CS-K-prototypes algorithm combines the CS algorithm and K-prototypes algorithm to further improve clustering and search performance. It assumes that the number of cuckoo nests is M, the maximum number of iterations is T, and the total time complexity of the CS-K-prototypes algorithm is $O[T \times (M + I \times N \times K + M \times N \times K) \times (D_n + D_c)]$. It can be seen that due to the introduction of CS algorithm for global optimization, the time complexity of CS-K-prototypes algorithm is significantly higher than that of K-means and K-prototypes algorithms. But this also brings better clustering and search performance, especially when dealing with complex datasets and high-dimensional data.

3.2 Construction of intelligent search model based on CS-K-prototypes algorithm

In the field of intelligent search in archives, traditional algorithms cannot accurately search for the required data due to the wide variety of archive data types and the inability to perform data clustering and classification [19-21]. The CS-K-prototypes algorithm can classify diverse types of data well and accurately search for the required data. Therefore, the study introduces the CS-K-prototypes algorithm for intelligent data search in archives to improve the accuracy of data search. The search process for archive data classification in the archives is shown in Figure 2.

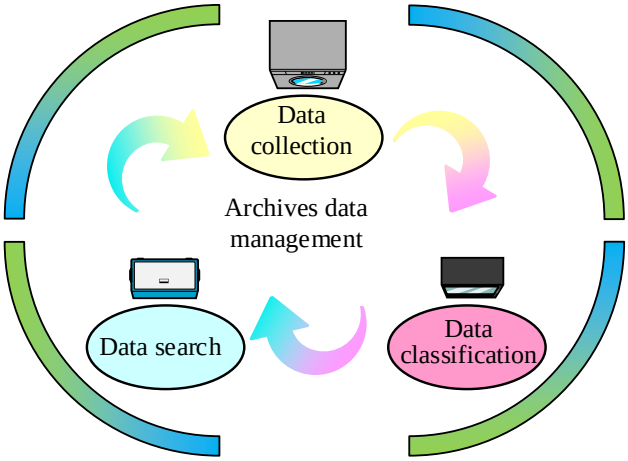


Figure 2: Archive data classification search process

In Figure 2, during data management, the archives collect archive data, classify the collected data by category, and finally search for the required archive data. The CS-K-prototypes algorithm first uses the CS-K-prototypes algorithm to classify the data in the archives for intelligent search. The data classification is mainly based on the distance of data attributes. The data attribute distance is used to measure the similarity between data points, providing a basis for clustering and classification. Its mathematical expression is shown in equation (9).

$$d(i,j)=\sqrt{\sum_{l=1}^{p}(X_{il}-X_{jl})} \quad (9)$$

In equation (9), i and j represent two different samples in the dataset, $d(i,j)$ represents the Euclidean distance between sample i and sample j, p represents the number of attributes in the data, and X_{i1} and X_{j1} represent the values of sample i and sample j on the l-th numerical attribute, respectively. Then, the dissimilarity of the data is calculated to evaluate the differences between data points and further optimize the clustering results, as shown in equation (10).

$$d(i,j)=\sum_{l=1}^{p}\left|X_{il}-X_{il}\right|^2+\gamma\sum_{l=p+1}^{m}\delta(X_{il},X_{il}) \quad (10)$$

In equation (10), m represents the number of attribute values in the data sample, and γ represents a constant weight. Subsequently, based on the dissimilarity of the data, the criterion function is calculated to comprehensively evaluate the dissimilarity of the data and guide the optimization direction of the algorithm. Its mathematical expression is shown in equation (11).

$$F(X,Q_i)=\sum_{i=1}^{k}\sum_{j=1}^{|x_i|}u_{ij}d_G(X_{ij},Q_i) \quad (11)$$

In equation (11), X_{ij} represents the attribute values in the dataset, G represents the subspace of the data, $|x_i|$ represents the quantity, Q_i represents the quantity of raw data, and $d_G(X_{ij},Q_i)$ represents the distance between different classification data. Finally, the spatial distance between different sample data is calculated based on the criterion function to ensure that the data points within each cluster are as similar as possible, further optimizing the clustering results. The calculation expression is shown in equation (12).

$$d_G(X_1,X_2)=\frac{d_{Gs}(X_1,X_2)+\eta d_{Gf}(X_1,X_2)}{|G|} \quad (12)$$

In equation (12), $d_G(X_1,X_2)$ represents the Euclidean distance between sample data, $\eta d_{Gf}(X_1,X_2)$ represents the overlap distance value between different sample data, η represents the weight value, and $|G|$ represents the number of dimensions of spatial distance between sample data. To better manage and classify the data in the archives, an intelligent search model for archive data based on the CS-K-prototypes algorithm is studied to achieve intelligent search management of archive data. The specific process of constructing the intelligent model using the CS-K-prototypes algorithm is shown in Figure 3.

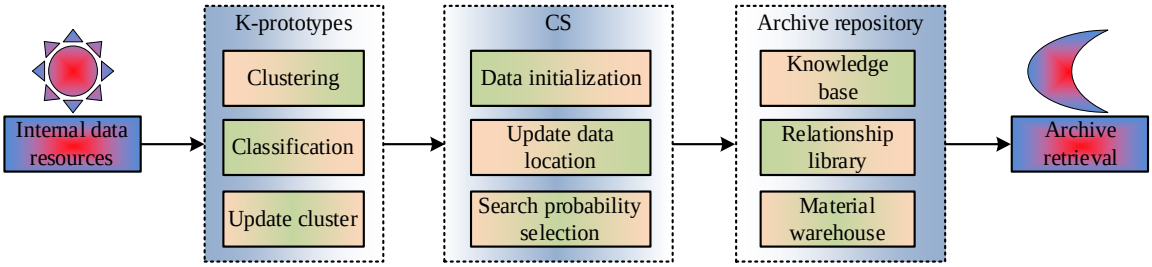


Figure 3: CS-K-prototypes algorithm intelligent search model

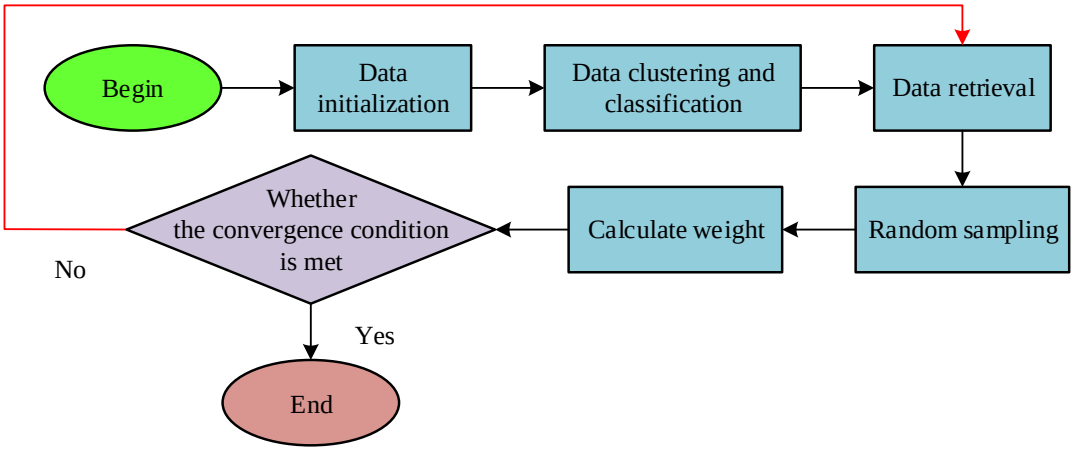


Figure 4: Using the CS-K-prototypes search model for data search in archives

In Figure 3, in the constructed CS-K-types search model, First, the data in the library were collected, and second, the CS algorithm was initialized, and the data location was updated. Finally, three types of databases were constructed for data retrieval. In archival data, knowledge bases are mainly used to store and manage unstructured or semi-structured knowledge data, which can flexibly handle the complexity and diversity of textual data, such as text archives, policy documents, meeting minutes, etc. The relational database uses a relational model to organize data and is suitable for storing structured archive data. It can accurately store and manage archive data with fixed formats, such as personnel files, accounting files, etc. The material warehouse is mainly used to store and manage archival data of multimedia or physical materials, such as image archives, audio archives, video archives, etc. When using the CS-K-prototypes search model for data search in archives, the data search process in the archives is shown in Figure 4.

In Figure 4, when CS-K-prototypes perform archive retrieval in the archive, they first initialize the data, cluster the initialized data, and classify the clustered data according to population attributes. Next, it randomly samples the data that has undergone classification operations, calculated the weights of the sampled data samples, and updated the data weights. Finally, it conducts data retrieval in the archive and ends the process.

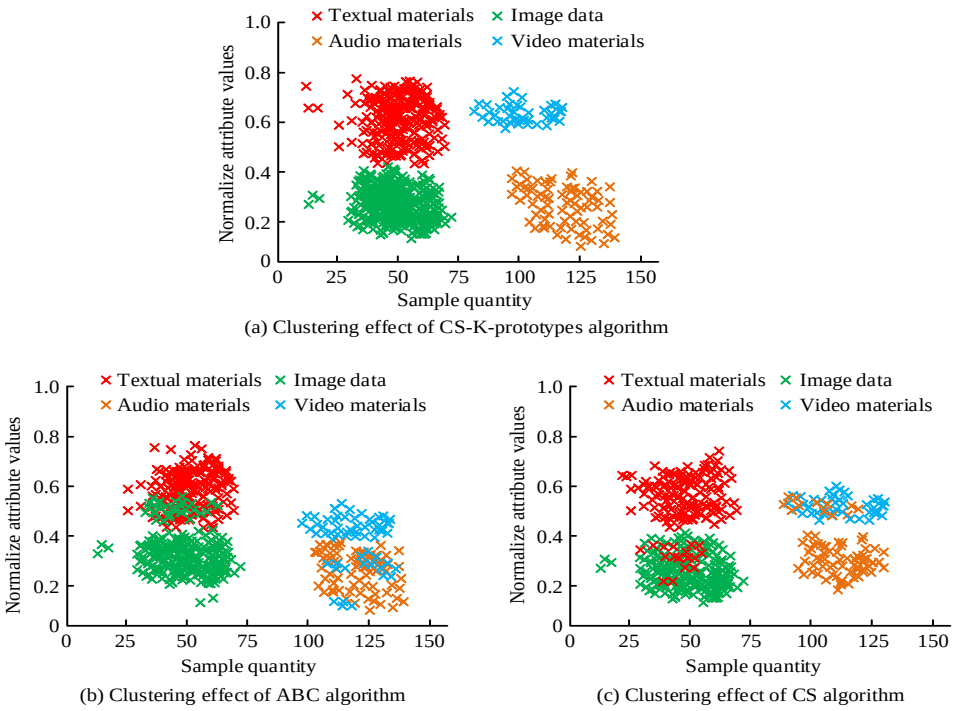
4 Empirical analysis of CS-K-prototypes search model

4.1 Performance verification based on CS-K-prototypes algorithm

To validate the effectiveness of the CS-K-prototypes algorithm proposed by the research, performance verification experiments were conducted using MATLAB software. The MATLAB software version was R2018a, the operating system was Windows 10, the graphics card was GTX 1650, the memory was 16GB, and the processor was Core i9-14900KS. The study collected 200 sets of data from public machine learning libraries to establish a dataset. These data included synthetic data and real-world data, covering various types of features, including numerical and subtyping attributes, to simulate the complexity and diversity of data in archives. The experimental set contained 150 sets of data for training and optimizing algorithm models. The test set contained 50 sets of data to validate the performance and generalization ability of the algorithm model. All data was preprocessed, including missing value padding, normalization, and one hot encoding, to ensure data consistency and quality. The number of bird nests was set to 25, the discovery probability was set to 0.25, the stride α was set to 1, β to 1.5, the iteration count was set to 500, and the step size factor was set to 1. The Artificial Bee Colony (ABC) algorithm and CS algorithm are optimization algorithms based on swarm intelligence, which exhibit good global search ability and convergence performance in solving optimization problems. Therefore, the study used ABC algorithm and CS algorithm as comparative algorithms. The population size of the ABC algorithm was set to 50, the maximum number of iterations was set to 100, and the probability of abandoning nests was set to 20. To represent the features of data points in two-dimensional space for clustering analysis and result visualization, this study normalized the data points and obtained normalized attribute values in the clustering space, ranging from 0 to 1. This ensures that the contributions of different attributes to the clustering results are relatively balanced, avoiding certain attributes dominating the clustering results due to their large range of values. The experimental results are shown in Figure 5.

Comparing Figures 5 (a), (b), and (c), regardless of whether the sample size was small (0-100) or large (100-150), the CS-K-prototypes algorithm could cluster different types of archival data into their respective regions more clearly when classifying different data, with less overlap between data points and more distinct

clustering boundaries. Both ABC algorithm and CS algorithm have experienced classification errors. The above results indicated that the CS-K-prototypes algorithm could better perform clustering and classification of different data compared to the other two algorithms. Precision reflects the accuracy of retrieval, while recall reflects the comprehensiveness of retrieval. The Precision-Recall (PR) curve can visually demonstrate the relationship between precision and recall, and comprehensively evaluate the retrieval performance of the algorithm at different thresholds. The K-means algorithm is one of the most classic and widely used clustering algorithms, characterized by simplicity and efficiency. Therefore, this study further incorporated the K-means algorithm as a comparative algorithm. The K value of the K-means algorithm was set to 3 and the convergence threshold was set to 0.0001. Comparison was conducted using 50, 100, and 150 sets of data, and the experimental results are shown in Figure 6.

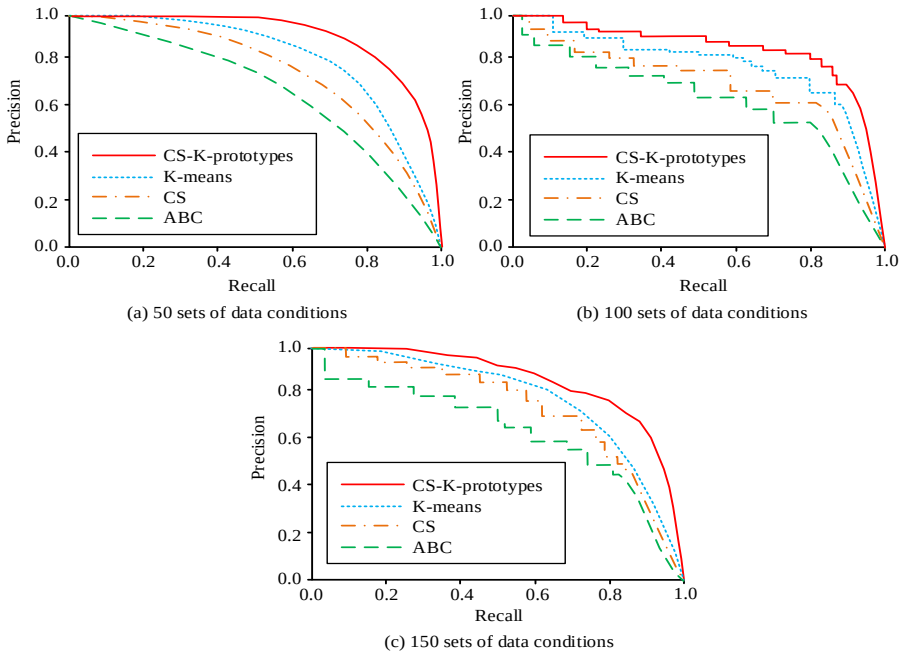


(a) Clustering effect of CS-K-prototypes algorithm

(b) Clustering effect of ABC algorithm

(c) Clustering effect of CS algorithm

Figure 5: Comparison of clustering effects of different algorithms



(a) 50 sets of data conditions

(b) 100 sets of data conditions

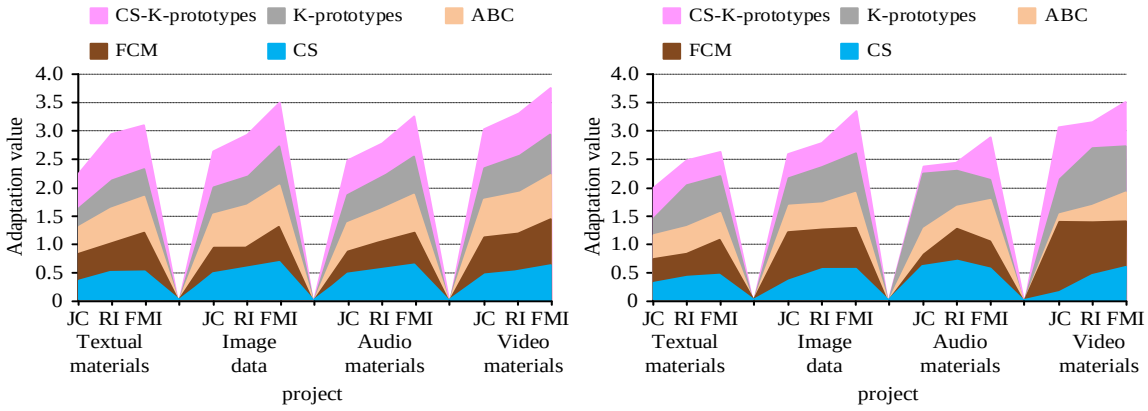
(c) 150 sets of data conditions

Figure 6: Recall rate variation results of different algorithms

According to Figure 6 (a), the PR curve offline area of the CS-K-prototypes algorithm was 0.9744, while the PR curve offline area of the K-means algorithm was 0.9014, low out of the study proposed model 0.0730. From Figure 6 (b), the offline area of the PR curve of the CS-K-prototypes model was 0.9678, while the offline area of the PR curve of the ABC algorithm was only 0.7426, which was 0.2252 lower than that of the CS-K-prototypes model. From Figure 6 (c), the offline areas of the PR curves of K-means algorithm and CS algorithm were close, with offline areas of 0.8847 and 0.8795, respectively. The above findings denoted that the CS-K-prototypes algorithm had higher search accuracy for data classification compared to the other three algorithms.

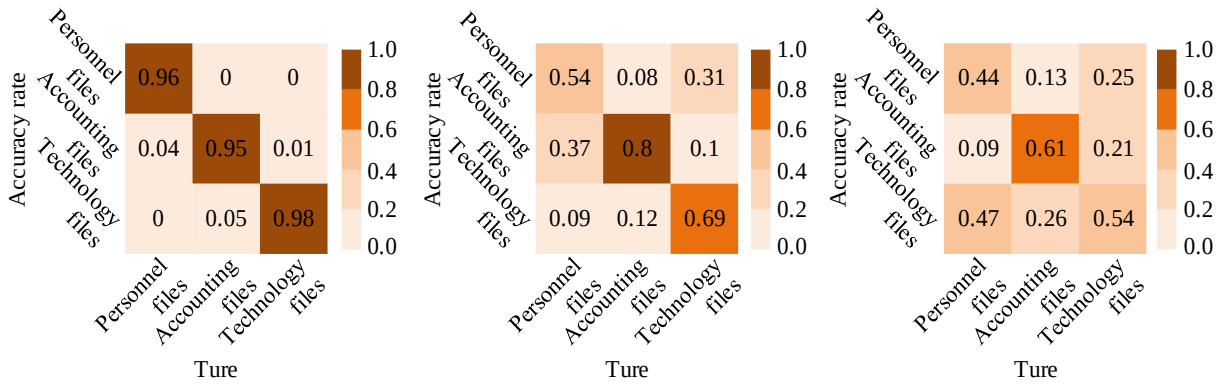
4.2 Analysis of the practical application effect of CS-K-prototypes search model

To analyze the practical application effect of the CS-K-prototypes search model, this study collected real-world archive data from the digital archive system of a large archive through the internet. These data cover different types of files, including 25 sets of personnel files, 50 sets of accounting files, 75 sets of technology files, 100 sets of text files, 50 sets of image files, 100 sets of audio files, and 150 sets of video files. The Jaccard Coefficient (JC) is used to measure the degree of overlap between search results and the actual set of relevant archives, and can intuitively reflect the accuracy of the algorithm in retrieving archive categories. The Rand Index (RI) evaluates the similarity between clustering results and real labels, and can comprehensively measure the accuracy of algorithms in classifying archival data. Fowlkes Mallows Scores (FMI) combines precision and recall to more fairly evaluate the algorithm's retrieval performance on different categories of archives. Therefore, this study selected JC, RI, and FMI as classification indicators. The Fuzzy C-Means Clustering (FCM) model was a classic fuzzy clustering algorithm that can handle the fuzziness and uncertainty of data. It has been widely used in fields such as image processing and text clustering, and can verify the superiority of CS-K-prototypes algorithm in data clustering. The CS-K-prototypes search model was compared with FCM model, K-means model, ABC model, and CS model. The fuzzy coefficient of the FCM model was set to 1.5, the convergence threshold was set to 0.0001, and the K value was set to 4. The experimental results are indicated in Figure 7.



(a) 100 sets of experimental data results (b) 200 sets of experimental data results

Figure 7: Classification performance of different algorithms



(a) CS-K-prototypes (b) BS (c) ES

Figure 8: Search results of three models on different archival data

According to Figure 7 (a), the CS-K-prototypes search model had a JC value of 0.47, RI value of 0.5, and FMI value of 0.52 when searching for text archives. Compared to the FCM model, the JC, RI, and FMI indicators decreased by 0.37, 0.45, and 0.77, respectively. According to Figure 7 (b), the CS-K-prototypes search model showed a decrease of 1.42, 1.13, and 1.28 in JC, RI, and FMI indicators compared to the ABC model when conducting image archive searches. The results indicated that the CS-K-prototypes model performed better in classifying and searching data in archives compared to other models. The search accuracy directly reflects the consistency between the archives retrieved by the algorithm and the user's needs, and can intuitively measure the algorithm's ability to accurately locate effective information from large-scale complex archive data. It is a key indicator for evaluating the practical application performance of the algorithm. The Binary Search (BS) model is a classic search algorithm with high search efficiency, which can evaluate the comprehensive performance of CS-K-prototypes algorithm in terms of search efficiency and accuracy. The Exponential Search (ES) model is an efficient search algorithm that performs well, especially when dealing with large-scale data. It can evaluate the efficiency and accuracy of the CS-K-prototypes algorithm when dealing with large-scale archival data. The CS-K-prototypes model was validated on the established dataset, search experiments were conducted on different data in the archive, and the CS-K-prototypes model was compared with the BS model and ES model for data search. The experimental outcomes are indicated in Figure 8.

From Figure 8 (a), the CS-K-prototypes model could accurately search for different archives, with a search accuracy of 96% for personnel archives, and a search accuracy of 95% and 98% for accounting and technology archives, respectively. In Figure 8 (b), the BS model had lower search accuracy when searching for different files, with a search accuracy of only 54% for personnel files, which was 42% lower than the CS-K-prototypes model. Moreover, when identifying accounting and technology files, its search accuracy was significantly lower than that of the CS-K-prototypes model. In Figure 8 (c), the ES model had significantly lower search accuracy than the other two models when searching for archives. The search accuracy for personnel archives was only 44%, which was 52% lower than the CS-K-prototypes model. However, when searching for accounting and technology archives, the search accuracy was only 61% and 54%, which was significantly lower than the CS-K-prototypes model. From this, among the three models, the CS-K-prototypes model can accurately search for archives in different archives. The proposed CS-K-prototypes model was further compared with three baseline models: Best Matching 25 (BM25), Word Frequency Inverse Document Frequency-based Vector Space Model (TF-IDF), and Document to Vector (Doc2Vec). Paired t-test was used, with a significance level set at 0.05, and the experiment was repeated 30 times. The accuracy comparison results of different models are shown in Table 2.

Table 2: Comparison of accuracy between different models

Archive data	BM25	TF-IDF	Doc2Vec	CS-K-prototypes
Personnel	78.38±2.42	72.11±2.96	81.54±2.07	96.02±0.77
Accounting	70.58±3.13	68.54±3.13	75.65±2.71	95.25±1.10
Technology	81.75±1.86	78.42±2.88	80.22±2.23	98.34±0.65
Text	82.24±4.25	80.54±2.15	80.57±2.09	97.54±0.73
Image	61.96±3.91	57.64±4.76	70.45±3.53	96.78±0.91
Audio	58.42±4.98	55.32±4.06	67.81±3.29	91.84±1.36
Video	63.59±3.67	60.14±3.99	72.60±3.04	93.19±1.21

From Table 2, the proposed CS-K-prototypes model had significantly higher search accuracy than the baseline models BM25, TF-IDF, and Doc2Vec in different types of archival data (p<0.001). This is because all baseline models require manual design of feature transformations, while CS-K-prototypes directly handle the original mixed attributes, reducing information loss. Although Doc2Vec improves semantic understanding through embedding learning, it is still limited in cross modal data.

5 Discussion

With the increasing diversity of data types in archives, aiming at the problem of low data search accuracy and poor classification performance of traditional search algorithms in archives, this study proposed using CS algorithm to improve K-prototypes clustering algorithm for intelligent search of archives. The K-prototypes algorithm can effectively handle clustering problems of mixed data, but there is a problem of insufficient search ability when facing large datasets and high-dimensional spatial data. The CS algorithm has the advantages of strong global search ability, fewer parameters, and faster convergence speed. Improving the K-prototypes algorithm through the CS algorithm can make it more efficient in exploring the solution space, avoiding getting stuck in local optima, and thus improving the search accuracy and efficiency of the intelligent search system for archives. Compared with traditional optimization methods such as particle swarm

optimization (PSO), genetic algorithm (GA), or ABC algorithm, CS-K-prototypes algorithm has certain trade-off advantages in computation. The CS algorithm has fewer parameters, faster convergence speed, and is easier to adjust and optimize during algorithm implementation, reducing the complexity and computational cost of parameter selection. The CS-K-prototypes algorithm can better maintain population diversity and global search ability when dealing with mixed and high-dimensional data, avoiding getting stuck in local optima.

The results showed that the CS-K-prototypes algorithm outperformed the ABC algorithm and CS algorithm in overall performance, and performed better in search accuracy and error control. The PR curve offline area of CS-K-prototypes algorithm could reach 0.9744, which was 0.073 higher than that of K-means algorithm. Under different dataset sizes and complexity conditions, the CS-K-prototypes algorithm also demonstrated good adaptability and robustness. When the dataset size was small (50 sets of data), the PR curve offline area of CS-K-prototypes algorithm was 0.9744, significantly higher than other algorithms. As the dataset size increased (100 and 150 sets), the R-curve offline areas of the CS-K-prototypes algorithm were 0.9678 and 0.9345, respectively, which still had significant advantages compared to other algorithms. In addition, in the practical analysis of the CS-K-prototypes model, the search accuracy of CS-K-prototypes for personnel files, accounting files, and technology files reached 96%, 95%, and 98%, respectively. The search accuracy of BS model for personnel files was only 54%, which was 42% lower than that of CS-K-prototypes model. The search accuracy of the other two types of file data was also significantly lower than that of CS-K-prototypes. The search accuracy of the ES model for personnel files was only 44%, which was 52% lower than the CS-K-prototypes model.

Compared with the testing scenario search method based on social cognitive optimization algorithm proposed by scholars such as Zhu, the proposed CS-K-prototypes algorithm focuses more on clustering, classification, and retrieval of data in archives, with stronger specificity. Although the improved cross modal retrieval method proposed by Geigle et al. performs well in cross modal retrieval, the CS-K-prototypes algorithm has superior performance in handling comprehensive retrieval of various types of data in archives. The text retrieval-based search method proposed by Liu et al. has high accuracy in text retrieval, but its retrieval performance is not as comprehensive as the CS-K-prototypes algorithm when facing the diverse data types in the archives. In addition, compared with the integrated clustering algorithm proposed by Li et al. and the meta heuristic optimization algorithm based on K-means algorithm proposed by Minh et al., the CS-K-prototypes algorithm has better clustering classification performance and search performance when dealing with mixed data, large datasets, and high-dimensional spatial data. The proposed CS-K-prototypes algorithm has strong clustering and search capabilities, and can handle both numerical and subtyping attributes simultaneously. It is not only suitable for archive data retrieval in archives, but also plays an excellent role in dealing with diverse document types such as images, audio, and video, and has broad generalizability. In practical applications, through appropriate preprocessing and parameter adjustment, the CS-K-prototypes algorithm can fully leverage its advantages and meet document retrieval needs in different scenarios.

6 Conclusion

This study proposed a CS-K-prototypes clustering algorithm that combines the advantages of K-prototypes algorithm in handling mixed data with the powerful global search capability of CS algorithm, aiming to improve the intelligent search performance of archive data. The experimental results demonstrated that the proposed CS-K-prototypes algorithm had high accuracy and good classification performance in intelligent search of archive data, which helped to solve the problems of insufficient search ability and unsatisfactory clustering effect of traditional clustering algorithms in processing archive data, and improve the retrieval efficiency of archive data. Not only applicable to archive data, but can also be extended to other document retrieval scenarios containing mixed data. However, the time complexity of CS-K-prototypes algorithm is relatively high, especially when dealing with large-scale datasets, which may face challenges in terms of computational efficiency. Moreover, the study only used archival data for testing, and the application effect of CS-K-prototypes algorithm in high-dimensional and time-series data types has not been verified yet. Therefore, in future work, parallel computing or distributed computing frameworks should be further adopted to accelerate the operation of algorithms, and how to combine CS-K-prototypes algorithms with other data processing techniques and models to address the challenges of different types of data.

References

[1] Poongodi M, Malviya M, Hamdi M, Vijayakumar V, Mohammed M A, Rauf H T, Al-Dhlan K. A. 5G based Blockchain network for authentic and ethical keyword search engine. IET Commun., 2022, 16(5): 442-448.

[2] Reiff S B, Schroeder A J, Kırlı K, Cosolo A, Bakker C, Mercado L, Park P J. The 4D Nucleome Data Portal as a resource for searching and visualizing curated nucleomics data. Nature communications, 2022, 13(1): 2365-2377.

[3] Tian C, Wan H, Wu Y. Fuzzy Similarity K-Type Prototype Algorithm and Marketing Methods.

Informatica, 2025, 49(13): 13-32.
[4] Mingyu Z, Sutong W, Yanzhang W, Dujuan W. An interpretable prediction method for university student academic crisis warning. Complex & Intelligent Systems, 2022, 8(1): 323-336.
[5] Zhu J, Ma X, Martínez L, Zhan J. A probabilistic linguistic three-way decision method with regret theory via fuzzy c-means clustering algorithm. IEEE Transactions on Fuzzy Systems, 2023, 31(8): 2821-2835.
[6] Qiao L, Li L, Yu S. Multi-Objective Optimization for Human Resource Allocation Using Reinforcement Learning and Enhanced Cuckoo Search Algorithm. Informatica, 2025, 49(19): 167-182.
[7] Mohammad O K J, Salih B M. Improving Task Scheduling In Cloud Datacenters By Implementation Of An Intelligent Scheduling Algorithm. Informatica, 2024, 48(10): 77-88.
[8] Oyewole G J, Thopil G A. Data clustering: application and trends. Artificial intelligence review, 2023, 56(7): 6439-6475.
[9] Wang J, Alroobaea R, Baqasah A M, Althobaiti A, Kansal L. Study on library management system based on data mining and clustering algorithm. Informatica, 2023, 46(9): 17-24.
[10] Amin F M, Rusydiyah E F, Azizah A N. Personalized Library Book Recommendations Using K-Means Clustering and Association Rules. Journal of Scientometric Research, 2025, 14(1): 32-45.
[11] Minh H L, Sang-To T, Wahab M A, Cuong-Le T. A new metaheuristic optimization based on K-means clustering algorithm and its application to structural damage identification. Knowledge-Based Systems, 2022, 251(6): 109-121.
[12] Zhang H, Li H, Chen N, Chen S, Liu J. Novel fuzzy clustering algorithm with variable multi-pixel fitting spatial information for image segmentation. Pattern Recognition, 2022, 121(6): 201-214.
[13] Shi Y, Zhang S, Wang S, Xie H, Feng J. Multiple-perspective consumer segmentation using improved weighted Fuzzy k-prototypes clustering and swarm intelligence algorithm for fresh apricot market. Italian Journal of Food Science, 2024, 36(4): 38-56.
[14] Hafid H, Annisa S. Implementation of k-medoids and k-prototypes clustering for early detection of hypertension disease. Barekeng: Jurnal Ilmu Matematika dan Terapan, 2025, 19(1): 465-476.
[15] Dong J, Wang Y, Chen X, Qu X, Li X, He Y, Wang X. Reading-strategy inspired visual representation learning for text-to-video retrieval. IEEE transactions on circuits and systems for video technology, 2022, 32(8): 5680-5694.
[16] Ikotun A M, Ezugwu A E, Abualigah L, Abuhaija B, Hemin, J. K-means clustering algorithms: A comprehensive review, variants analysis, and advances in the era of big data. Information Sciences, 2023, 622(4): 178-210.
[17] Yang L. Feature extraction of English semantic translation relying on graph regular knowledge recognition algorithm. Informatica, 2023, 47(8): 103-124.
[18] Li T, Rezaeipanah A, El Din E S M T. An ensemble agglomerative hierarchical clustering algorithm based on clusters clustering technique and the novel similarity measurement. Journal of King Saud University-Computer and Information Sciences, 2022, 34(6): 3828-3842.
[19] Awad F H, Hamad M M. Improved k-means clustering algorithm for big data based on distributed smartphone neural engine processor. Electronics, 2022, 11(6): 883-895.
[20] Feiyan Zhang. Thinking on The Information Construction and Standardized Management of University Construction Engineering Archives. Acta Informatica Malaysia. 2023; 7(2): 105-107. http//: doi.org/ 10.26480/aim.02.2023.105.107
[21] Liu C, Wang J, Zhou L, Rezaeipanah A. Solving the multi-objective problem of IoT service placement in fog computing using cuckoo search algorithm. Neural Processing Letters, 2022, 54(3): 1823-1854.