The Exploration and Application of K-medoids in Text Clustering

Qiongjie Dai[1,2], Jicheng Liu[1*]

[1] School of Economics and Management, North China Electric Power University, Beijing, China
[2] School of Mathematics and Computer Engineering, Ordos Institute of Technology, Ordos, Inner Mongolia, China
Email: daiqiongjie06041@163.com, ljc29@163.com

Abstract. Clustering algorithms is a statistical analysis method for classifying samples/indexes. The traditional text clustering algorithm is complicated and not convenient for data processing. Therefore, we proposed a new text clustering algorithm based on K-medoids. The new text clustering algorithm combines document category with semantics contribution. The new clustering algorithm can not only optimize the document frequency, but also take consideration of influence of the document category on the characteristic weight. The new text clustering algorithm was shown as follows: first, combine the proposed semantic contribution with fuzzy cluster, and vested the document (with no category information) category thereby; then we proposed the category information entropy and combined it with the semantic contribution in order to modify the traditional TF-IDF weight calculation method. We found the new text clustering algorithm was superior to the traditional weight calculation method after testing it in open platform of Chinese text categorization corpus data set. Therefore, we concluded that the new text clustering algorithm might have vast foreground of application.

To solve the shortcomings of the traditional weight calculation method of feature items, text clustering algorithm based on K-medoids was proposed. The frequency and inverse document frequency were improved, and the influence of document category on feature weight was further studied. At the same time, because there may not be any standard classification datasets in practice, a new weight calculation method combining category and semantic contribution was proposed. First, the semantic contribution was proposed and then combined with fuzzy clustering. A text set with category information was obtained by rough clustering of text set without category information. Then, the category information entropy was proposed and combined with the semantic contribution to improve the traditional TF-IDF weight calculation method. Thus, a more effective weight calculation method was obtained. The Chinese text categorization corpus dataset in open platform of Chinese natural language processing of Fudan University was used for testing. The results showed that the new method for weight calculation of feature items was superior to the traditional weight calculation method. It is concluded that the improved text clustering algorithm can be used in a wider range of occasions.

Keywords: K-medoids, XML document clustering, UCI dataset, cluster center

1 Introduction

Extensible markup language (XML) has become a common data representation and interchange format on Weh because of its universality, extensibility, usability, self-description, heterogeneity and development. With the explosive growth of the number of XML documents, people urgently need to acquire information knowledge from these documents. Automatic clustering of XML documents can not only enhance the organization of XML documents in the network, but also discover the links between unknown and implicit knowledge and documents from massive XML documents, which has important research significance. In addition to some text content, the XML document fund project also has the structural features of the element father node and the nesting of the descendant nodes. Therefore, the traditional document clustering algorithm is not suitable for the clustering of XML documents. At present, there are two division methods for clustering XML documents, including K-means and k-

[*] Corresponding author

medoids. The XML document data set contains a number of discrete objects, and the mean of K-means does not really reflect the actual situation of the whole cluster, and there is no practical significance. At the same time, the k-means algorithm is very sensitive to the outlier. In contrast, k-medoids uses a specific object as a cluster center, which solves the problem of K-means sensitivity to outliers. The k-medoids algorithm has the characteristics of simple partition and fast execution time. Its operation is also suitable for clustering XML documents. Therefore, k-medoids have been widely used in the clustering of XML documents.

Foreigner research of text clustering started in the 19th century. Nowadays, the technology is mature and is widely used in business decision-making. For example, retailers can acquire the consumer's consumption needs by collecting the information, then they specified the sale strategy after classifying and analyzing the category of the customer's consumption demand. The text clustering technology has been widely in other industries. Recently, more progress has been made since IBM developed the text clustering by TextMiner software. Previous research proposed [1] the eliminating redundant features by using unsupervised feature selection to process large databases. Combining the maximization algorithm with the feature extraction method, we can carry out feature selection and clustering at the same time [2]. Once the selection of the initial center point is selected, the initial cluster center point could be obtained by analyzing the label, which was obtained by analyzing the density distribution information of the data sample [3]. The reliability of clustering feature selection was enhanced after simplifying the set of feature items by the fuzzy rough set [4,5]. Domestic research on text clustering started in the 20th century, and it got rapid development and achieved good results since then. The feature weight could be improved by taking the common occurrence frequency of the word in consideration. Thereby, a certain number of representative feature items are selected to form a vector space model [6].

2 Selection of Text Clustering Algorithm

Clustering and classification are two different forms of things division. Cluster analysis is a process of making a collection of physical or abstract sets into multiple classes made up of similar objects. In other words, the target object is automatically grouped in the absence of pre-class tagging information. The grouping process is to divide the target object into multiple categories according to a certain distance scale. The clustering result is to ensure that the objects in the same category have a very high similarity, while the objects in the different category need to have a very low similarity. Text clustering is the process of dividing a set of unordered sets of text into multiple groups or multiple categories by using a specified clustering method. Moreover, the text in the same group or in the same class has a very high similarity after the division.

Text clustering algorithm is a complex and unsupervised machine learning method. The clustering result will be directly influenced by the selection of clustering algorithms. Commonly used text clustering algorithms are divided into partitioning method, hierarchical method, density method, grid method and model method. The partitioning method is a clustering algorithm which is widely applied in clustering algorithm. The algorithm optimization discussed in this paper is based on pollen clustering algorithm.

The goal of clustering algorithm for partition is to classify a data sample in the data set to the corresponding K class clusters, and a class cluster can represent a category. It usually provides a certain way to give K initial cluster centers before dividing the cluster. Then, the remaining data samples are sorted into the corresponding categories according to the similarity. Therefore, the initial class cluster is formed. Next, we need to determine whether the division results are as expected. If the target function is not expected, the data samples should be re-divided again according to the iterative repositioning technique [7,8]. The correlation degree of data sample in the same cluster is relative high compared to in different clusters, and the correlation degree of the data set in the different class clusters is in the least level. The algorithm terminates can final K data collection. The process was shown in figure 1.

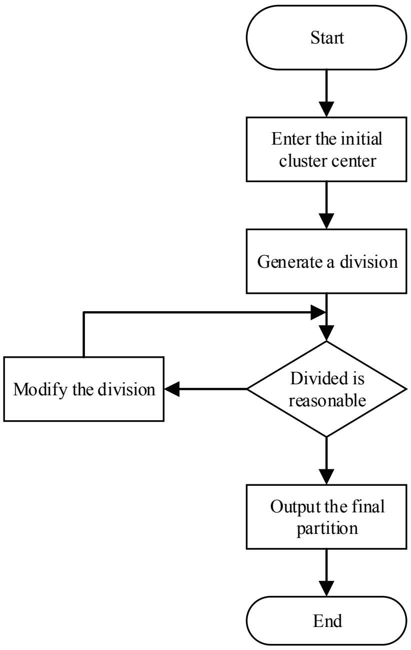


Figure 1. Clustering algorithm flow based on partition

Up to date, many clustering algorithms have been put forward, such as K-means, EM, K-medoids, PAM, CLARA, CLARANS and so on. They can be roughly divided into two categories based on the center point and the gravity point. Among them, the partition clustering algorithm based on the gravity point is not only a specific data sample but also present the average level of the data sample in the cluster. The algorithm is relatively simple, so that it can be widely used in clustering. The representative algorithm of this class is K-means. A specific data sample was set as the representative point of the class cluster during the partition clustering algorithm, which was based on the center point. After the representative point was close to the center of the cluster, it would be hard to be affected by the data noise, which was far away from the cluster data. Thus, the influence of the isolated data sample in the clustering was avoided. The representative algorithm of this class was K-medoids [9].

Dempster and other researchers proposed the EM algorithm in 1977. The algorithm introduced the probability and statistics knowledge on the basis of K-means. The data samples were divided according to their membership degree. All of the algorithms could be divided into two steps: E and M. The E step was used to estimate the expected value of the current data sample, and the M step gave the expected value to the unknown data sample. E and M were repeated alternately until they could convergence.

PAM was an algorithm based on K-medoids. The algorithm analyzed two combinations of all samples during the collection. The representative clustering results were calculated by using one of them in the combination. In an iterative process, if the clustering quality was improved, the representative was changed to another data sample and then it came into the next iteration. The selected representative object would be the center point of the clustering if the clustering mached the expectation of the objective function. However, the algorithm also had some disadvantages, and it was very complicated.

K-medoids and PAM algorithm can use repeated iterations to cluster until the representative point of the cluster center was accurate. Therefore, the algorithm had strong robustness. However, according to the clustering process of the algorithm, the two algorithms also had their own defects: their high complexities. Therefore, it was not suitable for clustering large-scale data sample. However, if the initial clustering center was selected, a better representative sample point could be selected too, which can greatly reduce the complexity of the clustering algorithm. Therefore, the CLARA algorithm was proposed. The algorithm did not choose the representative point randomly from the data set, but chose the data sample with the representative category point through the PAM algorithm. Then, the representative points could be used to cluster the whole data set. The number of samples that could be processed by this algorithm was bigger than the PAM. However, the sample might be affected by the clustering. Therefore, the clustering effect of this clustering algorithm depended on the selection of the initial center point [10].

The CLARANS algorithm was proposed on the basis of CLARA. The biggest difference between this algorithm and CLARA was to select a data sample with randomness during the iteration. This method avoided the defect that the best representative data sample could not be the best center point due to the limited selection range during the CLARA algorithm, which could affect the clustering. This kind of clustering input was relatively better. The clustering quality and the scalability were improved. But it also had its own shortcomings: the low computation efficiency and the sensitivity to the sequence of the input data. The algorithm could cluster the convex or spherical boundary [11].

3 Optimization of K-medoids Clustering Algorithm

K-medoids clustering algorithm was one of the classical algorithms based on partition, and it had been widely used because of its low sensitivity to noise. However, K center point clustering algorithm also had some shortcomings: the algorithm was very complex, and it could not determine the appropriate number of clusters in advance [12]. It had no uniform clustering evaluation criteria function, and the initial center point might be wrong selected.

3.1 Algorithm Optimization Path

To ensure the initial clustering center could be located in different clusters, a new method based on the distribution characteristics was proposed to optimize the local variance and the neighborhood radius. Considering the overall distribution, the Num value was determined by the overall radius, and the global optimal solution was obtained as much as possible. At the same time, because each local sample distribution was different, the number of sample points outside the same radius could be chosen as the Num value of each sample. Therefore, different sample distribution characteristics would get different Num values. The local variance and neighborhood radius was redefined. The method could dynamically calculate the neighborhood radius of the corresponding sample point, and selected a better initial cluster center point by establishing the neighborhood radius.

The algorithm idea came from probability and mathematical statistics. If there was a large fluctuation among the data distribution and the average number, the variance would be larger. As the same way, if there was a small fluctuation among the data distribution and the average number, the variance would be smaller. According to the definition, the variance would be smaller when it was located in a region or central area with more centralized data distributed. To rationalize the selection of the initial cluster center, the K initial center points were within the K cluster, and the center of the K cluster should be as much as possible. This ensured that the initial cluster center was located in the dense area of the samples. But the initial cluster center must be in different clusters. The detailed steps of the algorithm to select the initial cluster center were as follows:

Step 1: The number of initial partition numbers K, the instance data set D and the nearest sample parameter Nun were input;

Step 2: The local variance $F(x_i)$ of each data object x_i was computed. According to the value of $F(x_i)$, the samples in the data set X were arranged in an ascending order. The sample set X' was obtained. Then, the point set M of initialization cluster center would be empty, that was $M=\{\}$;

Step 3: The initial partition center of the class cluster was selected. The first sample value x_1 from the sample set X' were named as the initial partition center of a class cluster. Then, adding it to point set M of initial cluster center, that was, $M = M \cup \{x'_1\}$. After that, the object was removed from the data set, $X = X \cup \{x'_1\}$;

Step 4: According to the formula (4), the radius of the sample was calculated firstly, and the neighborhood $neigh(x_1)$ of the sample x'_1 was calculated next: $X' = X' - neigh(x'_1)$;

Step 5: it was necessary to return to the third step until the elements of the cluster center point set M was k, that was, $|M| = K$;

Step 6: The initial center point set M was output.

The algorithm used the uniform Num value after defining the local variance and neighborhood radius of the sample. The influence of the local distribution characteristics of the sample on the initial center selection was considered. Compared with the traditional clustering algorithm, the clustering effect was better. The Num value could be used to calculate the neighborhood radius of the sample point, which might result in two cases. First, the radius of the neighborhood was too large, and the center point which was selected should be deleted. But this would affect the selection of the initial center point. The second was that the neighborhood radius was too short, and the deleted sample point was not deleted and selected as the next central point. This would affect the selection of the initial center point, and the operation could be more complicated by artificially selecting the Num experience value.

3.2 Algorithm Process Selection

To solve the initial cluster center selection, the global optimum and local optimum must be considered. Therefore, a method was proposed to optimize the local variance and the neighborhood radius according to the distribution characteristics of each sample point. And by this, a better initial cluster center point could be obtained.

The relevant definitions were as follows:

The Radius value was defined as:

$$Radius = \frac{1}{n(n-1)} \times \sum d(x_i, x_j) \tag{1}$$

The *Num* value of the number of sample points around the sample x_i was defined as:

$$Num = num[a(x_i - x_j) > t \times Rddius] \tag{2}$$

The parameter t was a radius adjustment coefficient with the step length of 1 (from 0 to 10).

The local variance of the sample x_i was defined as:

$$F(x_i) = \frac{\sum_{j=1}^{Num}\left[d(i,j) - \frac{\sum_{j=1}^{Num} d(i,j)}{Num}\right]^2}{Num - 1} \tag{3}$$

The neighborhood radius of the sample x_i was defined as:

$$L_1 = \sqrt{F(x_i)} \tag{4}$$

$$neigh(x_i) = \{x_1 \mid d(i,l) \le L_1; l = 1,2,\ldots,n\} \tag{5}$$

The algorithm was divided into four stages, and the detailed process was shown in figure 2.

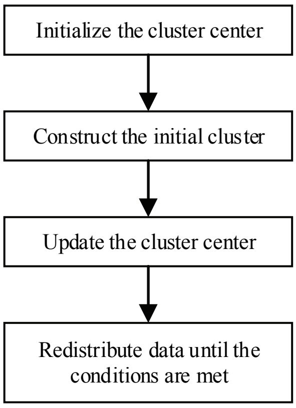

Initialize the cluster center
Construct the initial cluster
Update the cluster center
Redistribute data until the conditions are met

Figure 2. Algorithm process framework

The detailed procedures for each step of the algorithm were as follows:

Initialization cluster center:

Step 1: The average distance between all sample points was calculated according to the formula (1), and the cluster center point set M was null, that was, $M=\{\}$;

Step 2: In each sample point, the $t \times Radius$ was used as a radius, and the Num value of the n sample points was calculated according to the formula (2). The following steps were as follows:

Setting $Num=0$;

The distance d_{ij} from d_j to d_i was calculated, $j=(1,2,...,n)$;

If $d_{ij} > t \times Radius$, then $Num{+}{+}$;

Step 3: The local variance of each sample point was calculated according to the formula (3) and the neighborhood radius of each sample point was obtained according to the formula (4);

Step 4: The sample point with the minimum local variance was taken as the initial cluster center point and added to the set M;

Step 5: Sample points within the center and its neighborhood radius were deleted according to the formula (4) and (5);

Step 6: The steps from 2 to 5 were repeated until the initial center point set M contained K center points;

Step 7: The center point set M was output.

The construction of initialization cluster:

Step 1: All the sample points of the data set were assigned to the center point close to M, and the initial class division was obtained;

Step 2: Calculate the sum of squared error of initial clustering partition.

The cluster center point was updated:

Step 1: Calculate the new center of each cluster, and minimize the error square sum of the other data samples in the new center to the class cluster;

Step 2: Update the center point of all class clusters.

The redistribution of data:

Step 1: All the sample points of the dataset were assigned to the center point close to M;

Step 2: The sum of square error of the clustering error was calculated. If the ratio did not change, the algorithm ended. Otherwise, we should to return to step c to continue the operation.

3.3 Machine Learning Analysis of Algorithm

To test the clustering performance of the algorithm, the below equipments were need: the classic dataset of UCI machine learning database, the simulated dataset containing different scale and different proportions are used to do the experiment. The experiment used a high-performance computer with Pentium (R) Dua-Core E5800 3.20 GHz CPU, 48G memory, 500G hard disk, Win7 64-bit operation system. In addition, the Java language was used to implement algorithm in the Myeclipse 10.0 development environment. The modified K-medoids algorithm was compared with the traditional K-medoids algorithm as follows:

Five common clustering evaluation indexes were used in the performance evaluation of the algorithm, including clustering error square sum, RI index, precision, recall and F1 value. If TP was a sample point of the same class, TP would be divided into the same cluster, and TN was the sample point of different classes, TN would be divided into different clusters. If FP was a sample of different classes, it would be divided into the same cluster, and if TN was the same class of samples, so TN would be divided into different clusters.

$$RI = (TP + TN) / (TP + FP + FN + TN) \tag{6}$$

$$Precision = TP \; / \; (TP + FP) \tag{7}$$

$$Recall = TP \, / \, (TP + FN) \tag{8}$$

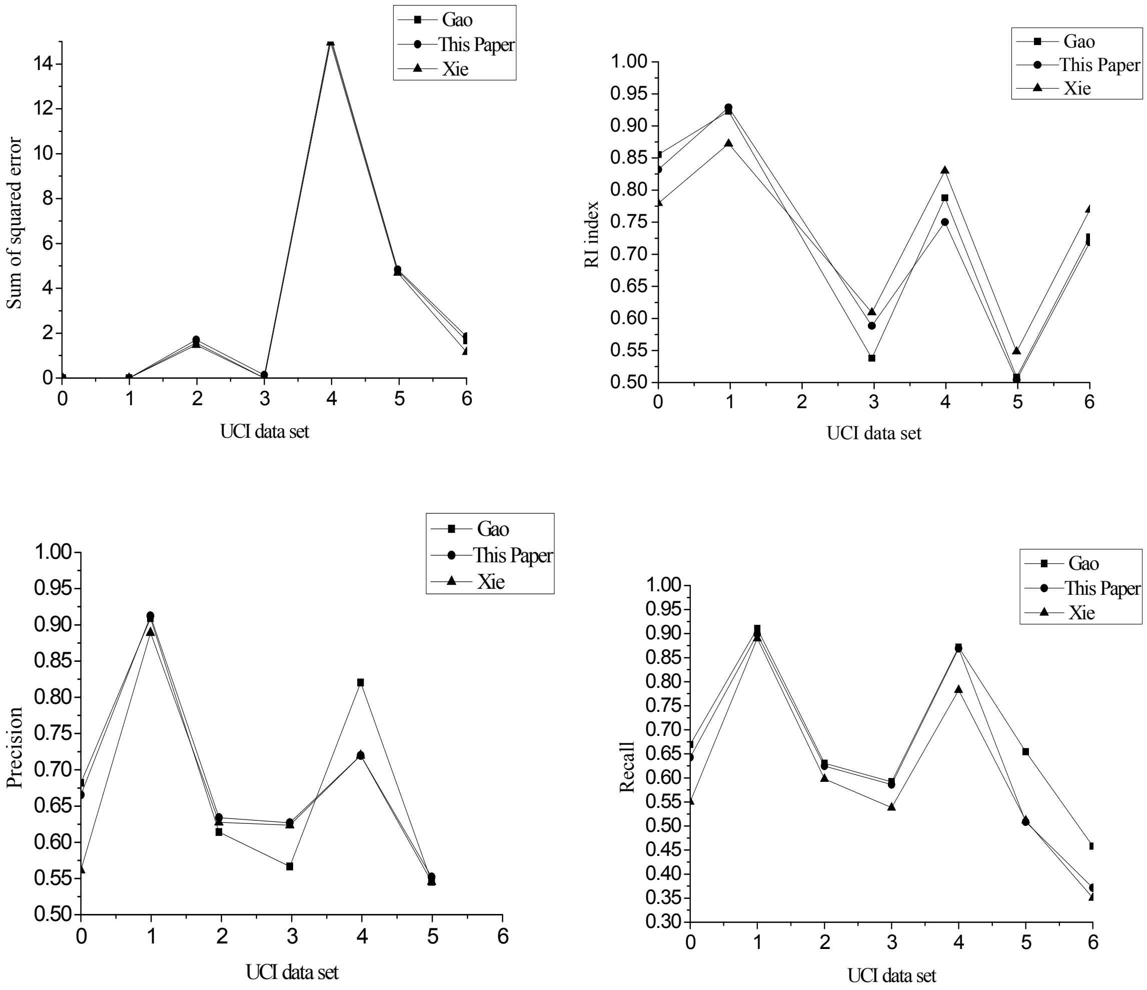
$$F1 = 2 \times Recall \times Precision \, / \, (Recall + Precision) \tag{9}$$

The 10 classic test clustering algorithms of UCI machine learning database were commonly used as data sets Segmentation, Wine, Yeast, Soybean, Iris and so on. Among them, the Soybean dataset selected Soybean-small, and Segmentation selected a large data set contained 2310 samples. Table 1 described the data set used in detail.

Table 1. Data set description and corresponding T values

Data source	Dataset marking	Sample number	Attribute number	Category number	*t*
Soybean-small	a	56	42	5	4
Iris	b	180	5	4	2
Wine	c	214	16	4	4
Ionosphere	d	421	41	2	1
WDBC	e	683	36	2	7
Pimalndians-Diabetes	f	922	10	2	2
Segmentation	g	2475	171	27	26

Figure 3 showed the test evaluation indexes of different algorithms on different data sets. The experimental results showed that the other indexes in data set 2 (Iris) were lower than the other two algorithms. The Precision index in data set 3 (Wine) was slightly smaller than the other two algorithms. However, the remaining five indexes were better than the Xie and Gaos' algorithm. In data set 1 (Soybean-small), the Recall index was smaller than the others. The other five indexes were better than the Xie and Gao's algorithm. Among the remained four data sets, the five indicators were much better than the other two algorithms.



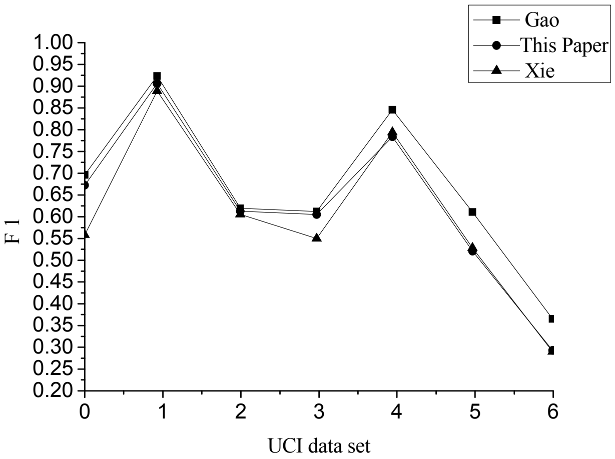


Figure 3. Clustering results of UCI data sets

The above experimental resulted in UCI dataset showed that the modified algorithm was better than the traditional algorithm, which effectively improved the clustering and the scalability of the algorithm.

As the improper initial center selection existed in the K-medoids clustering algorithm (which was based on partitioned clustering algorithm) had an impact on the clustering results, a radius adaptive K-medoids clustering algorithm was proposed. The idea and process of the algorithm were analyzed. We compared the experimental results of the algorithm.

4 Practical Application Test of Improved Algorithm in Text Processing

The experiment platform of Chinese text clustering system designed in this paper was introduced, and the related experiments were designed to compare the modified methods, which were proposed in this paper. To verify the effectiveness of the modified method, we compared the general evaluation of the clustering effect with the standard accuracy and the recall rate.

4.1 Application Test Text Source and Clustering Process

The algorithm was tested by the Chinese text classification corpus test in Fudan University. The corpus included environment (200), traffic (214), military (249), education (220), a computer (200), medicine (204) and sports (450), (505), political art (248), economic (325). The text randomly selects 200 sets of environment and computer in the dataset, 200 sets of education, medicine and art.

4.2 Analysis of Experimental Results of Algorithm

The text clustering system of Chinese text consists of three modules: pre-processing module, text representation module and clustering module. The pre-processing module included Chinese text segmentation and removal of stop word processing. The text representation module included fuzzy clustering module and feature weight calculation module. The fuzzy clustering module included feature extraction and weight calculation module. The clustering module contained the clustering algorithm before and after improvement. The specific process was shown in figure 4:

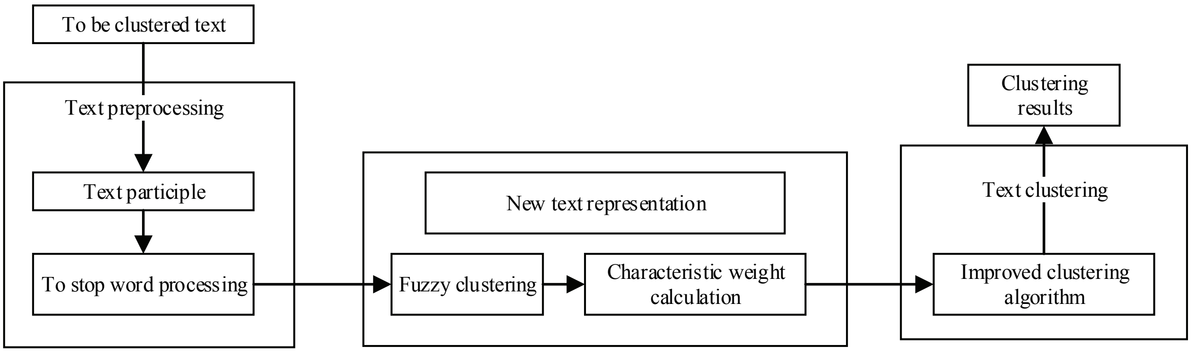


Figure 4. Flow chart of Chinese text clustering system

4.3 Analysis of Cluster Experiment Result

To compare the effect of algorithm improvement, the Chinese text categorization corpus data in Fudan University was used. The text randomly selected 200 sets of environment and computer in the dataset, 200 sets of education, medicine and art. The effect was evaluated by the accuracy and recall rate of the general evaluation index of clustering method. After comparing the experimental results between traditional K-medoids clustering algorithm (We considered the category effect and traditional K-medoids clustering algorithm, and ignored the category effect), the experimental results were shown in Table 2.

Table 2. Comparison of the experimental results between traditional K-medoids clustering algorithm

	Project	Environment	Computer	PE	Medicine	Art
Traditional K-medoids clustering algorithm without considering the category effect	Number of clustering texts	221	233	247	254	245
	Number of correct texts	186	210	214	230	214
	Accuracy rate	0.84	0.90	0.87	0.90	0.87
	Recall rate	0.705	0.795	0.81	0.87	0.81
Traditional K-medoids clustering algorithm considering the category effect	Number of clustering texts	227	242	246	241	244
	Number of correct texts	197	220	218	231	222
	Accuracy rate	0.87	0.91	0.89	0.96	0.91
	Recall rate	0.745	0.835	0.825	0.875	0.84

The comparison of the experimental results between modified K-medoids clustering algorithm (we consided the category effect, the modified K-medoids clustering algorithm, and ignored the category effect) was shown in Table 3.

Table 3. Comparison of the experimental results between improved K-medoids clustering algorithm t

	Project	Environment	Computer	PE	Medicine	Art
Improved K-medoids clustering algorithm without considering the category effect	Number of clustering texts	224	236	246	251	242
	Number of correct texts	197	219	223	238	220
	Accuracy rate	0.88	0.93	0.91	0.95	0.91
	Recall rate	0.75	0.83	0.85	0.90	0.84
Improved K-medoids clustering algorithm considering the category effect	Number of clustering texts	232	241	244	242	241
	Number of correct texts	210	227	228	239	223
	Accuracy rate	0.91	0.94	0.94	0.99	0.92
	Recall rate	0.80	0.86	0.87	0.91	0.85

The experimental results show that the modified K-medoids clustering algorithm was better than the traditional K-medoids clustering algorithm.

5 Conclusion

As inappropriate initial center point selection of K-medoids clustering algorithm would affect the clustering result, a radius adaptive K-medoids clustering algorithm was proposed. The idea and process of the algorithm were analyzed. We also compared the experimental results of the algorithm. The experimental results showed that the clustering results of the improved K-medoids clustering algorithm were better than the traditional K-medoids clustering algorithm.

In the process of this study, some problems need further consideration. For example, in the K-medoids clustering algorithm, the K value of the cluster type is provided firstly. It is necessary to find a way to judge the number of categories automatically.

Acknowledgements. This research was funded by National Natural Science Foundation of China (Grant number: 71771085), Scientific Research Project of Universities in Inner Mongolia (Grant number: NJZC17415) and Research Project of Ordos Institute of Technology (Grant number: KYYB2017013).

References

1. Han, J., Sun, Z., & Hao, H. (2015). L 0 -norm based structural sparse least square regression for feature selection. Pattern Recognition, 48(12), 3927-3940.
2. Xu, J., Liu, J., Yin, J., & Sun, C. (2016). A multi-label feature extraction algorithm via maximizing feature variance and feature-label dependence simultaneously. Knowledge-Based Systems, 98(C), 172-184.
3. Rathborne, J. M., Longmore, S. N., Jackson, J. M., Kruijssen, J. M. D., Alves, J. F., & Bally, J., et al. (2015). A cluster in the making: alma reveals the initial conditions for high-mass cluster formation. Astrophysical Journal, 802(2).
4. Peker, M. (2016). A decision support system to improve medical diagnosis using a combination of k-medoids clustering based attribute weighting and svm. Journal of Medical Systems, 40(5), 1-16.
5. Broin, P. Ó., Smith, T. J., & Golden, A. A. (2015). Alignment-free clustering of transcription factor binding motifs using a genetic-k-medoids approach. BMC Bioinformatics., 16(1), 1-12.
6. Mojahed, A., & Iglesia, B. D. L. (2017). An adaptive version of k -medoids to deal with the uncertainty in clustering heterogeneous data using an intermediary fusion approach. Knowledge & Information Systems, 50(1), 1-26.
7. Zhao, X., Li, Y., & Zhao, Q. (2015). Mahalanobis distance based on fuzzy clustering algorithm for image segmentation. Digital Signal Processing, 43(C), 8-16.
8. Abin, A. A., & Beigy, H. (2015). Active constrained fuzzy clustering: a multiple kernels learning approach. Pattern Recognition, 48(3), 953-967.
9. Ferreira, C. S., Lachos, V. H., & Bolfarine, H. (2016). Likelihood-based inference for multivariate skew scale mixtures of normal distributions. Asta Advances in Statistical Analysis, 100(4), 1-21.
10. Mandur, J. S., & Budman, H. M. (2015). Robust algorithms for simultaneous model identification and optimization in the presence of model-plant mismatch. Industrial & Engineering Chemistry Research, 18(12), 1470-1481.
11. Velmurugan, T. (2018). A state of art analysis of telecommunication data by k-means and k-medoids clustering algorithms. Journal of Computer & Communications, 06(1), 190-202.
12. Khatami, A., Mirghasemi, S., Khosravi, A., Lim, C. P., & Nahavandi, S. (2017). A new k-medoids clustering and swarm intelligence approach to fire flame detection. Expert Systems with Applications, 68(C), 69-80.