Available online at www.ijpe-online.com

International Journal of Performability Engineering

vol. 15, no. 3, March 2019, pp. 756-762
DOI: 10.23940/ijpe.19.03.p4.756762

Clustering Algorithm of Ethnic Cultural Resources based on Spark

Ming Lei[a,b], Bin Wen[a,*], Jianhou Gan[b], and Jun Wang[b]

[a]*School of Information Science and Technology, Yunnan Normal University, Kunming, 650500, China*
[b]*Key Laboratory of Educational Informatization for Nationalities of Ministry of Education, Yunnan Normal University, Kunming, 650500, China*

Abstract

Extracting valuable information from ethnic cultural resources is the key to current data mining research on ethnic cultural resources. The K-means algorithm can effectively process large-scale data sets due to simple and efficient iterative calculations. The uncertainty of the k-value affects the efficiency and accuracy of the algorithm. The particle swarm optimization (PSO) algorithm and global coarse-grained search can quickly determine the k-value of the cluster center, while the retrieval efficiency is low. In order to solve the problem of the initial clustering center of the K-means algorithm and the low efficiency of the PSO algorithm, this paper proposes a Spark-based PSO-k-means algorithm, which primarily introduces ethnic cultural text resources into the Hadoop Distributed File System (HDFS) and then uses Han Language Processing (HanLP) word segmentation. The Term Frequency-Inverse Document Frequency (TF-IDF) algorithm generates the word frequency vector. Finally, the particle swarm optimization algorithm performs initial pre-clustering on the data set, obtains the K-means algorithm cluster center k, and then obtains the final classification result through K-means algorithm cluster analysis. The experimental results show that the clustering accuracy and stability of the PSO-k-means algorithm are better than those of the existing K-means algorithm on serial stand-alone.

Keywords: ethnic culture; particle swarm optimization algorithm; K-means clustering

(Submitted on October 19, 2018; Revised on November 21, 2018; Accepted on December 23, 2018)

1. Introduction

At this stage, ethnic cultural resources and main means of communication are concentrated on ethnic cultural websites, government websites, or local forums in ethnic areas. Due to the large number of ethnic groups in China, the diversity of ethnic cultures and their unstructured characteristics make knowledge difficult to be exploited. The content of ethnic cultural resources is also scattered and large, and traditional data mining algorithms cannot be effectively processed. Meanwhile, there are frequent inconsistencies in the description and understanding of ethnic cultures on different websites and platforms. The mutual relationship between knowledge cannot be figured out, which leads to ambiguity or differences in people's understanding of ethnic cultures. It hinders people's correct cognition and understanding of ethnic culture and affects the inheritance of ethnic culture.

In the field of data mining of ethnic cultural data resources, K-means has a fast convergence speed and simple implementation in the data mining process. However, in the processing of massive data, it is vulnerable to the problem of initial cluster center uncertainty and data storage under stand-alone conditions. A k-means algorithm that satisfies differential privacy under the MapReduce cloud platform was proposed by Li et al. [1] to ensure better usability when improving privacy and timeliness. Applying the particle swarm optimization algorithm to the selection of the initial clustering center of the data effectively solves the problem of the clustering center of the K-means algorithm, but it does not improve the processing efficiency of the data [2-3]. An adaptive cuckoo K-means clustering algorithm for the clustering algorithm was proposed by Wang et al. [4] to fall into the optimal problem. On the Hadoop platform, the accuracy and efficiency of the algorithm are improved. The SCoS algorithm was proposed by Zhu et al. [5] for the large spectrum algorithm, and the Apache Spark parallel computing framework adopts multiple rounds of iterative methods through matrix sparseness. The representation and storage of the SCoS algorithm has high computational performance and good data scalability in large-scale data sets.

* Corresponding author.
E-mail address: wenbin@ynnu.edu.cn

In the MapReduce model, through the combination of multiple algorithms, the canopy algorithm was introduced to initialize the clustering center of the K-means algorithm, significantly improving the stability and accuracy of the algorithm [6]. Li proposed fuzzy classification of data sets through the idea of fuzzy clustering [7] and then classified the data set twice through dynamic computing of the clustering center. The combination of various algorithms improves the speedup of the algorithm, and the convergence speed becomes faster. A grid-based privacy protection clustering data mining method was proposed by Cui et al. [8]. They constructed a privacy data security homomorphic algorithm by lattice encryption, improving the accuracy and efficiency of the algorithm. An unsupervised algorithm mines texts in online medical forums, which can effectively predict changes in adverse reaction labels [9]. On MapReduce, the distributed forest model of the large-scale data set is distributed and classified by introducing a random forest model with information gain of input feature weighting coefficients [10].

The distributed parallelization framework Spark not only solves the storage problem of massive data, but also improves the efficiency of data mining calculation through parallelization. The preprocessing of national text data resources is performed. Data preprocessing refers to the process of data fusion, removal of stop words, and normalization of the previous original text resource data, and it plays a decisive role in the efficiency of the data mining process algorithm. The PSO-k-means algorithm is used to implement data mining in Spark programming, extracting valuable knowledge and relationships among different ethnic groups. Applying PSO-k-means on Spark not only solves the clustering center of the K-means algorithm, but also effectively improves the accuracy, stability, and efficiency of the algorithm.

2. Related Work

Spark is a generic parallel open source framework similar to Hadoop MapReduce, which was developed by the UC Berkeley AMP lab [11]. Spark inherits all the functions of MapReduce. However, Spark is different from MapReduce because the former is based on the memory-based elastic distributed data set. The calculation intermediate output and the final result output can be stored in the memory to improve the interactive query efficiency of the data. It is unnecessary for Spark to read and write HDFS files repeatedly, saving Input and Output (IO) overhead between the disk and memory and balancing iterative work. It is very suitable for data analysis and machine learning algorithms involving large amounts of data through iterative calculation of streaming read [12]. The core of Spark is based on the directed no-cyclic graph (DAG) computing model and Resilient Distributed Datasets (RDD). RDD is a read-only object of distributed partition. It automatically splits the data into slices in the cluster. The machine data set joint partition control realizes the unity of data. Data can be reconstructed to achieve automatic fault tolerance and ensure data integrity [13]. DAG is the dependency between multiple stages generated by Spark after submitting the job. The data is moved with DGA in the process and is divided into different stages by shuffle. The programming interface operations provided by different stages RDD are also different. Transformation operations, such as map, filter, filterMap, sort, reduceByKey, etc. return an RDD data set [14]. In the Transformation stage, the lazy strategy is adopted.

The various conversion operations are not executed immediately. Execution is triggered only after the operator in the Action stage has been submitted, such as count (), collect (), and save (). The execution is triggered to reduce various conversions. The intermediate data that needs to be stored in the operation is better in real-time processing, as shown in Figure 1. Compared with other parallel frameworks, RDD has better fault tolerance and more flexible data exchange. It successfully builds an integrated and diversified big data system, which can greatly improve the efficiency of massive data parallel processing.

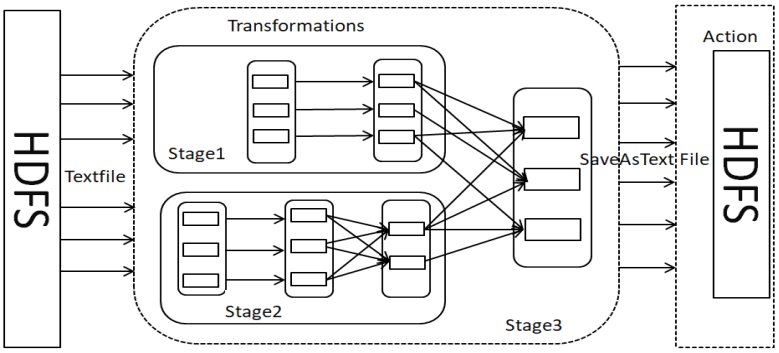

Figure 1. Transformation and action of RDD

3. Research and Implementation of Related Algorithms

3.1. K-Means Algorithm

K-means is a distance-based clustering algorithm [15]. The similarity of clusters and nodes is evaluated by the size of the comparison distance. The distance between the particles is small, indicating that they are close to each other and representing they are the same class, and the particles have a high degree of familiarity. Firstly, the K-means algorithm randomly selects k objects in the dataset as the initial clustering center and then calculates the distance of each object to each cluster center, puts the object into the cluster with the smallest distance, and repeats the iteration. The cluster centers of each type are recalculated until the cluster centers of the k clusters no longer change or the clustering function converges, and then the clustering is completed. In the data processing stage, it is assumed that k objects are randomly selected from the n-dimensional data set $X=\{x_1,x_2,\cdots,x_n\}$ as the cluster center, and the cluster center set is $Z=\{z_1,z_2,\cdots,z_k\}$. The distance of each object to each cluster center is evaluated using the measure of Euclidean distance $D=(x_i,z_h)$, $(i=1,2,\cdots,n;\ h=1,2,\cdots,k)$ of each object to each cluster center. If $D=(x_i,z_h)=\min_{x\in X,z\in Z}\|x-z\|$, the data object is merged into the class. After the data is clustered, the new cluster center is recalculated according to the adjusted classification, as shown in Equation (1):

$$C_h^*=\frac{1}{n_h}\sum_{i=1}^{n_h}x_i^h \tag{1}$$

Where n_h is the number of objects in the subset after classification, and then the square sum criterion function J is calculated, as shown in Equation (2):

$$J^*=\sum_{k=1}^{n_h}\sum_{h=1}^{k}\|\,x_k^h-c_h^*\,\|^2 \tag{2}$$

If for any $\left|J^*-J\right|<\delta$, it means that the clustering criterion function converges and the final clustering result is obtained; otherwise, it is still necessary to continue to iterate until the iteration is jumped, and the maximum iteration number ensuring the algorithm can also be set has good robustness.

3.2. Particle Swarm Optimization Algorithm

The Particle Swarm Optimization (PSO) algorithm is a new type of swarm intelligence evolution algorithm inspired by famous scholars Kennedy and Eberhart in the late 20th century based on the phenomenon of finding food routes during the foraging of birds [16]. During the search process, the particles update their position and speed by tracking the current optimal particles. A set of random particles is iterated by a certain number of steps, and any iteration depends on the current two extreme values to obtain the optimal solution, which is very suitable for complex nonlinear problems that are difficult to solve by traditional algorithms. The individual extremum is the local optimal solution found by the particle in the iterative process, as shown in Equation (3):

$$Pbest_i=(Pbest_{i1},Pbest_{i2},Pbest_{i3},\cdots,Pbest_{iq}) \tag{3}$$

The global extremum is the optimal solution found by the population of all particles in the iterative process, which is shown in Equation (4):

$$Gbest_i=(Gbest_{i1},Gbest_{i2},Gbest_{i3},\cdots,Gbest_{iq}) \tag{4}$$

In the case where the particle population is m particle groups and the particle search dimension is q, the velocity and position of the particle are both q-dimensional vectors, and the q-dimensional particle position is expressed as $X_i=\{x_{i1},x_{i2},\cdots,x_{iq}\}$. q-dimensional particle speed can be expressed as $V_i=\{v_{i1},v_{i2},\cdots,v_{iq}\}$, and the particle movement function is shown in Equations (5)-(6):

$$V_i(t+1) = w \times V_i(t) + c_1 \times r_1 \times (Pbest_i - X_i) + c_2 \times r_2 \times (Gbest_i - X_i) \quad (5)$$

$$X_i(t+1) = X_i(t) + V_i(t+1) \quad (6)$$

$Pbest_i$ represents the individual optimal position of particle i, $Gbest_i$ represents the optimal group position of particle i, $V_i(t)$ and $X_i(t)$ respectively represent the velocity and position of particle i at the t times iteration, and the speed at which the particle is iterated by the speed of the last iteration is denoted by w. The individual optimal position of the particle and the influence of the optimal position of the group on the particle velocity are respectively expressed as c_1 and c_2, and their value is 2; r_1 and r_2 are random weights defining the particle velocity uniformly distributed in the interval (0, 1) [17]. Suppose $f(x)$ is the fitness function of the particle swarm, the fitness value corresponding to the individual optimal position is $Pbest_i$, the group position $Gbest_i$ is called the individual extremum $f(Pbest_i)$, and the group extremum is $f(Gbest_i)$. If the current fitness value of the particle is better than the individual extreme value, the current fitness value is $f(Pbest_i) = f(x_i)$, and the optimal position of the particle is replaced by the optimal position of the current particle. If the fitness value of the (t + 1) times particle is better than the current fitness value, the fitness value updates by $Pbest_i(t+1) = x_i(t+1)$; otherwise, the current fitness value is not updated. The global optimal value of the particle swarm is also called the global fitness value, that is, the optimal value of the individual fitness value $f(Pbest_i)$ in all particle swarms.

3.3. Spark-based Parallel PSO-K-Means Algorithm

When performing data clustering, the computationally efficient and simple K-means algorithm is easy to fall into the local optimal solution, and the initial clustering center is very important for the clustering results [18]. Many experts and scholars combine the optimized particle swarm optimization algorithm with the K-means algorithm to improve the initial clustering center of the K-means algorithm and improve the accuracy and stability of the algorithm [19]. Data parallel computing can be handled by the big data parallel computing framework MapReduce but reading HDFS data and other external data to memory every time, batch processing, IO operation, and computing phase communication are all highly time-consuming. Spark realizes the real-time processing of data by batch processing and interactive query in memory. Spark calls the RDD operator instead of MapReduce's Map and Reduce operations, and the processing efficiency is significantly better than that of Hadoop MapReduce [20]. The partitioning and merging of data is the key of Spark. The data is allocated to different nodes according to a fixed size, and each node is calculated and then totaled to the master node. The improved PSO-k-means algorithm based on Spark is composed of two parts: PSO and K-means. Firstly, input the data set N and the number of clusters k. Secondly, the particle swarm algorithm is used to continuously perform iterative calculation on the particles to find the global optimal position, which is k in the initial cluster of the K-means algorithm. Finally, the K-means algorithm uses random sampling data to process the fragments, performs the combined clustering process after one completion followed by constant iteration calculation, and then finally obtains the k best clustering clusters. The parallel algorithm of the PSO-k-means algorithm based on Spark platform is described as follows:

Algorithm 1: PSO-k-means algorithm

Input: number of clusters *k*, raw data *N*, number of populations of particle population *m*

Output: *k* group clustering results

1. Generate a class cluster center file that records the class cluster center for each iteration.
2. Initialize particle position, velocity, and *k* clusters.
3. Repeat.
4. Record the particle position, velocity, and *k* clusters into the cluster center file and broadcast to each node.
5. Each node calculates its own data and the cluster center, generates data corresponding to the data points, and writes them to the cluster center file.
6. The cluster center updates based on the collected information for the next iteration.
7. Exit the loop if the cluster center has not changed.
8. Enter the *k* cluster centers that are iteratively updated.
9. Initially divide *k* cluster centers.
10. Repeat.
11. Record *k* clusters into the cluster center file and broadcast to each node.
12. Each node calculates its own data and the cluster center, generates data corresponding to the data points, and writes them to the cluster center file.
13. The cluster center updates based on the information collected for the next iteration.
14. Exit the loop if the cluster center has not changed.
15. **end**

The data object in the PSO stage is subjected to the adaptive cache management strategy [21], and the reused automatic cache RDD is selected, which can be reused multiple times in the subsequent K-means clustering. Spark will automatically realize the parallelization of the PSO-k-means clustering algorithm according to the logic of the algorithm. The task assignment and data distribution in the parallelization implementation are automatically implemented by the system. The parallel flow of the PSO-k-means algorithm is shown in Figure 2.

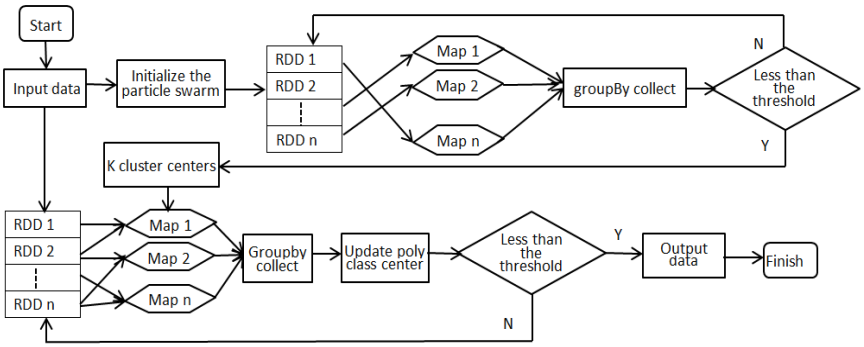

Figure 2. PSO-k-means parallel flow chart

4. Experiment and Analysis of Results

4.1. Experimental Environment

This experiment is based on the requirements of the Key Laboratory of the Ministry of Education and Information Technology and basic parallel computing requirements for Spark configuration. It consists of three Lenovo desktop workstations, equipped with CentOS 7 system, CPU quad-core 3.5GHz, memory 8G, and cluster environment spark2.3.0. The running mode is Spark on Yarn and the corresponding Hadoop version is 2.7.5, consisting of one master node and two slave nodes. The network address configuration is shown in Figure 3. The NameNode is responsible for managing and allocating resources to the DataNode.

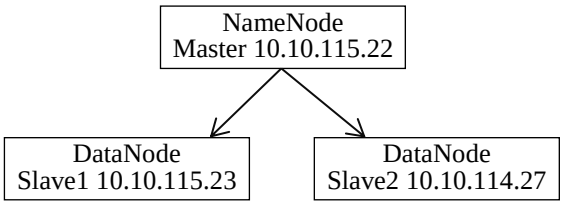

Figure 3. Cluster node distribution

4.2. Experimental Analysis

In order to verify the effectiveness of the improved algorithm, the accuracy of the algorithm is verified. The data used is the Key Laboratory data of the Ministry of Education, and the data is divided into groups A, B, and C. The data size of group A, B, and C is 10Kb, 100Kb, and 1MB respectively. Over the course of the experiment, the three sets of data sets were run 20 times and averaged as the final experimental result data to reduce the influence of random errors. Set the number of iterations of the algorithm to 50 and the number of particles to 12. The operation results are shown in Table 1.

Table 1. Accuracy of K-means and PSO-k-means algorithms %

Data set	K-means		PSO-k-means	
	Serial	parallel	Serial	Parallel
A	74.12	76.24	81.45	82.26
B	77.35	78.46	82.65	83.44
C	76.48	77.34	83.54	85.35

It can be seen from Table 1 that the four algorithms are used to calculate the data sets A, B, and C respectively. Since K-means is sensitive to the initial cluster center and the accuracy is low, the PSO algorithm can preferably determine the initial aggregation of the K-means algorithm. At the class center, the accuracy of the algorithm is higher and more stable. The accuracy of the K-means algorithm is lower than the accuracy of the improved PSO-k-means algorithm, as shown in Figure 4.

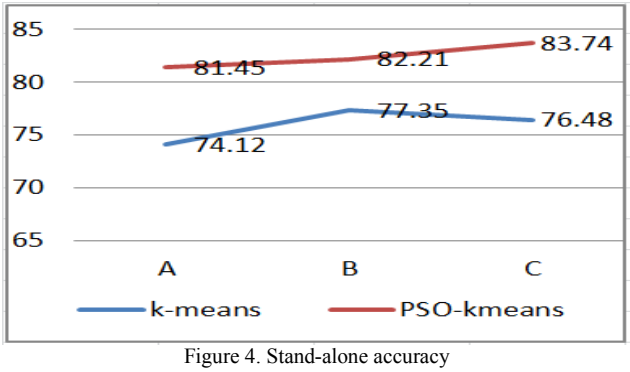

Figure 4. Stand-alone accuracy

Furthermore, the parallel PSO-k-means algorithm has higher accuracy for data operations than the serial PSO-k-means algorithm. As the amount of data grows larger, the advantages of the parallel PSO-k-means algorithm become more prominent, as shown in Figure 5. Therefore, the parallel PSO-k-means algorithm proposed in this paper has higher accuracy and better stability than the traditional K-means on Spark.

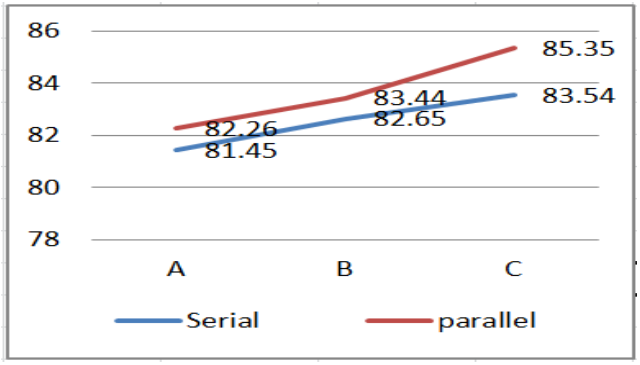

Figure 5. PSO-k-means accuracy

5. Conclusions

The text data of massive ethnic cultural resources is taken in this paper as the research object. For the traditional K-means algorithm, the initial clustering center is uncertain, resulting in low accuracy and insufficient stability. The global search ability of PSO algorithm is good, but the speed is slow. Based on the analysis of the characteristics of the two algorithms, a Spark-based PSO-k-means algorithm is proposed to form the word frequency vector in the distributed environment. The initial pre-clustering process of the PSO algorithm is used to obtain the cluster k-value of the classified data set. Finally, the K-means algorithm achieves accurate classification of data sets. The accuracy and efficiency of implementing the PSO-k-means and K-means algorithms in a single-machine and cluster environment are compared and analyzed. The experimental results show that the PSO-k-means algorithm significantly reduces the number of iterations in the K-means phase, which is steadier. At the same time, the PSO-k-means algorithm has shorter running time and higher efficiency in Spark. Next, we will study larger data and the number of clusters and verify whether the algorithm has good validity and scalability when dealing with larger scale data. The effects of fewer categories of particles on classification results will also be further studied.

Acknowledgements

This research is supported by the National Nature Science Fund Project (No. 61562093, 61661051), Key Project of Applied Basic Research Program of Yunnan Province (No. 2016FA024), Program for Innovative Research Team (in Science and Technology) in University of Yunnan Province, Application Infrastructure Projects of Science and Technology Plan in Yunnan Province (No. 2016FD022), and Starting Foundation for Doctoral Research of Yunnan Normal University (No. 2017ZB013).

References

1. H. C. Li, X. P. Wu, and Y. Chen, “K-Means Clustering Method Supporting Differential Privacy Protection under MapReduce Framework,” *Journal on Communications*, Vol. 37, No. 2, pp. 124-130, 2016

2. A. Bolfazlis, S. Anaeiz, and A. Hmede, "Cloud-based Augmentation for Mobile Devices: Motivation, Taxonomies, and Open challenges," *IEEE Communications Surveys and Tutorials*, Vol. 16, No. 1, pp. 337-368, 2014
3. Y. Shen, D. H. Yu, and W. L. Wang, "Improvement of Particle Swarm K-means Clustering Algorithm," *Computer Engineering and Applications*, Vol. 50, No. 21, pp. 125-128, 2014
4. B. Wang and X. J. Yu, "Parallel K-Means Clustering Algorithm for Adaptive Cuckoo Search," *Application Research of Computers*, Vol. 3503, pp. 675-679, 2018
5. G. H. Zhu, S. B. Huang, C. F. Yuan, and Y. H. Huang, "SCoS: Design and Implementation of Parallel Spectral Clustering Algorithm based on Spark," *Chinese Journal of Computers*, Vol. 41, No. 4, pp. 868-885, 2018
6. X. Y. Li, L. Y. Yu, H. Lei, and X. F. Tang, "A Parallel Implementation and Application of an Improved K-Means Algorithm," *Journal of University of Electronic Science and Technology of China*, Vol. 4601, pp. 61-68, 2017
7. L. Y. Li, Y. M. Dong, and Y. Kong, "Improved MapReduce Parallelization of K-Means Algorithm," *Journal of Harbin University of Science and Technology*, pp. 31-35, 2016
8. Y. H. Cui, W. Song, Z. B. Wang, S. C. Shi, and F. Q. Cheng, "A Grid-based Privacy Protection Clustering Data Mining Method," *Journal of Software*, Vol. 28, No. 9, pp. 2293-2308, 2017
9. R. Feldman, O. Netzer, and B. Rosenfeld, "Utilizing Text Mining on Online Medical Forums to Predict Label Change due to Adverse Drug Reactions," in *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, pp. 1779-1788, 2015
10. F. Qiao, Y. Z. Ge, and W. C. Kong, "Research on Distributed Improvement of Random Forest Student Employment Data Classification Model based on MapReduce," *Systems Engineering - Theory & Practice*, Vol. 37, No. 5, pp. 1383-1392, 2017
11. K. Sun, "Research and Implementation of Machine Learning Application Framework based on Spark," Shanghai Jiaotong University, 2015
12. P. Cao, "Optimization and Implementation of Clustering Algorithm based on Spark Platform," Beijing Jiaotong University, 2016
13. B. Zhang, "Parallelization and Optimization of K-Means Algorithm based on Spark," Huazhong University of Science and Technology, 2015
14. Y. Liang, "Parallelization of Data Mining Algorithms based on Distributed Platforms Spark and YARN," Sun Yat-Sen University, 2014
15. Y. H. Zhang and F. G. Li, "Parallelization of KMeans Clustering Algorithm based on MapReduce," *Journal of Jiujiang University* (*Natural Science Edition*), pp. 73-75, 2017
16. Y. Yang, S. X. Ren, J. Yan, and C. Q. Li, "Improved Log-based Optimization based on K-Means Algorithm for Web Log Mining," *Journal of Computer Applications*, Vol. 36, No. S1, pp. 29-32+36, 2016
17. D. F. Wang and L. Meng, "Performance Analysis and Parameter Selection of Particle Swarm Optimization Algorithm," *Acta Automatica Sinica*, Vol. 42, No. 10, pp. 1552-1561, 2016
18. Y. N. Liao, M. J. Li, and Y. Q. Zhang, "K-Means Clustering-Particle Swarm Optimization Multi-Target Localization Algorithm," *Electronic Design Engineering*, Vol. 26, No. 2, pp. 56-60, 2018
19. X. X. Lin and M. X. Zhao, "A K-Means Algorithm based on Improved Particle Swarm Optimization Algorithm," *Journal of Shandong University of Technology* (*Natural Science*), Vol. 29, No. 5, pp. 16-20, 2015
20. X. D. Wu and S. Q. Qi, "Comparison of MapReduce and Spark for Big Data Analysis," *Journal of Software*, 2018
21. C. Bian, W. Yu, and C. T. Ying, "Adaptive Cache Management Strategy for Parallel Computing Framework Spark," *Chinese Journal of Electronics*, Vol. 45, No. 2, pp. 24-30, 2017

Ming Lei is a Master's student in the School of Information Science and Technology at Yunnan Normal University. His research interests include machine learning and data mining.

Bin Wen received his Ph.D. in computer application technology from China University of Mining & Technology in 2013. In 2005, he was a faculty member at Yunnan Normal University. Currently, he is an associate professor at Yunnan Normal University. His research interest covers intelligent information processing and emergency management.

Jianhou Gan received his Ph.D. in metallurgical physical chemistry from Kunming University of Science and Technology in 2016. In 1998, he was a faculty member at Yunnan Normal University. Currently, he is a professor at Yunnan Normal University. His research interests cover education informalization for nationalities, Semantic Web, databases, and intelligent information processing.

Jun Wang received his Master's degree in modern education technology from Yunnan Normal University in 2012. In 2013, he was a faculty member at Yunnan Normal University. Currently, he is an assistant research fellow at Yunnan Normal University. His research interests cover education informalization for nationalities and knowledge engineering.