Send Orders for Reprints to reprints@benthamscience.ae

288 *The Open Cybernetics & Systemics Journal*, 2015, *9*, 288-294

Open Access

Research on K-Means Algorithm Based on Parallel Improving and Applying

Deng Zhenrong[1,2,*], Deng Xing[1], Zhang Chuan[1], Xu Liang[1] and Huang Wenming[1,2]

[1]*School of Computer Science and Engineering, Guilin University of Electronic Technology, Guilin, 541004, P.R. China;*
[2]*Guangxi Key Laboratory of Trusted Software, Guilin, 541004, P.R. China*

Abstract: The capacity of single server or CPU is unable to finish the task of the mining of mass data. In consideration of this bottleneck problem, a combined algorithm which is used by genetic and MR-based parallel clustering algorithm is proposed. To make up for the defects of clustering analysis in screening the clustering center, the clusters are used by genetic algorithm, relying on M-R parallel computing model to accelerate the convergence of the clustering analysis. To verify reasonableness of algorithm, this algorithm which is applied to analysis of the actual log is based on building of Hadoop platform. Experimental results show that relying on performance of distributed cluster computing and genetic clustering analysis to process log files can get better clustering results, and the efficiency of mining of massive log can be greatly improved.

Keywords: Cloud computing, Clustering analysis, Genetic algorithm, Map-reduce, Mass data processing.

1. INTRODUCTION

Nowadays, the arrival of the era of big data is not in doubt. The data showed the trend of rapid growth in scientific research (Astronomy, biology, high-energy physics), the Internet, e-commerce, and computer simulation application. Especially, the new amount of data generated in the annual scientific research is about 15PB; the era of big data has two major trends: the data expansion and the depth analysis of data. Eighty percent data is unstructured data in some large enterprises. This grows exponentially every year. Large data challenge not only involves the construction of enterprise storage architecture, data center infrastructure, but can also produce chain reactions for cloud computing, data mining, business intelligence, data warehouse, etc. Enterprises invest more energy in business analysis and for mining the TB level data in the future, for example, study of mining in massive log and log depth analysis will be an essential part of [1]. The log is a very broad concept term in the computing world; any program may have a log file, such as the application of computer kernel system, computer server, all kinds of businesses and social networking sites. The log file contains the information of interest, as the business enterprise develops the potential value of commercial information by analysis of user link and statistics such as the number of access in the logs. Therefore, analysis and mining of a log have become a hot topic in computer research field. Various characteristics of large data such as outstanding?, have made traditional log processing, log mining methods and algorithms no longer applicable. Facing the intensive, complex hybrid log file needs to have a more efficient method of computing and date mining algorithm.

*Address correspondence to this author at the School of Computer Science and Engineering, Guilin University of Electronic Technology, Guilin, 541004, P.R. China; Tel: 13977315675; E-mail: zhrdeng@guet.edu.cn

This paper is based on parallel computing model of MapReduce that has caused widespread attention [2, 3]. The combined algorithm [4-6], which is used by genetic and MR-based parallel clustering algorithm is proposed in view of the advantages of genetic algorithm. The algorithm follows the thought of sample global optimization, task of divide and rule, and results of summary and analysis. On one hand, it solves the defect of cluster center initialization instability for k-means algorithm, and on the other hand, accelerates the convergence of K-Means algorithm. The algorithm is deployed on the Hadoop experimental platform, verifying the effectiveness of the algorithm.

2. ALGORITHM ANALYSIS

2.1. Genetic Algorithm

Genetic algorithm is a general learning method based on evolution. Genetic algorithm is no longer as other algorithms searching hypothesis ranging from general to specific and from simple to complex, but selects the parents sample through a certain probability from the initial sample groups. The genetic algorithm can control the global search process, finally obtaining the search optimal solution set or near optimal solution set. Compared with the traditional heuristic search, genetic algorithm advantage is the colony search strategy. The main work is the choice of genetic operation parameters which can be completed from frequent human-computer interaction process in the search process.

2.2. K-Means

Clustering analysis is that the data set is divided into family, the family data are similar, and different data as far as possible. It can mine valuable information distribution patterns in the data and is an important means of mass information processing. It also plays a vital role in machine learning, data mining, text analysis and other fields. Classi-

1874-110X/15

cal clustering algorithm is compared to the K-Means clustering, fuzzy clustering and spectral clustering. the log of servers and custom transaction data, similar customer groups, web pages clustering, and clustering algorithm of frequent access path are analyzed [7] based on coupling matrix processing. Distributed affinity propagation clustering algorithm based on MapReduce is further discussed [8]; this algorithm overcomes the limitation of the dense datanot Sparse. [9, 10] Three clustering methods are clustered: clustering method based on group, clustering method based on granularity, and clustering method based on fuzzy. These algorithms cannot reflect the advantage in big data analysis.

K-Means is a clustering algorithm based on iterative distance. Its realization is simple, the efficiency is higher for large data processing, especially, it gets better when mining high dimension and agglomerate data. The K-Means algorithm observes instance classification to K clustering where it is less than the other clustering center distance. The K-means algorithm consists of three steps:

(1) Find the cluster center initialization, K is defined as the number of clusters;

(2) Calculate the distance for each observation instances to the cluster center, while putting instances to the nearest cluster. Distance uses Euclidean distance criterion. The calculation formula is as follows:

$$D(X_i, Y_j) = \sqrt{\sum_{j=1}^{n} (X_{ij} - Y_{ij})^2}, i = 1,2,...,n; j = 1,2,...,k \quad (1)$$

(3) Calculate the average distance of each cluster for all observed instances, and the average value as a new clustering center. The calculation formula is as follows:

$$center_i = \frac{1}{|s_i|} \sum_{x_j \in s_i}^{n} x_j \quad (2)$$

Repeat the third step until the clustering center no longer changes. Terminate the clustering process when achieving the optimal objective function or a maximum number of iterations. When using the Euclidean distance as the metric, the calculation formula is as follows:

$$\min \sum_{i=1}^{K} \sum_{x \in C_i}^{n} dist(c_i, x)^2 \quad (3)$$

2.3. MapReduce

Map-Reduce is an excellent model of distributed computing. It is widely applied to log analysis, mass data sorting, and also in massive data search. Business logic is abstracted into two functions by Map-Reduce in the large scale cluster system: Map and Reduce. Usually, the input data is divided into several independent data blocks by Map-Reduce. Processing the data block is parallel to Map function, and calculation model of frame is to sort the results of the Map task. Then the results are outputted to the Reduce function. Its core idea is to work (through the mapping) and Reduce (simplification). Each phase of the Map-Reduce process is as follows:

(1) Input: The input data is divided into several independent data blocks in this phase.

(2) Map: The user input data is regarded as several groups such as <key, value> key by Map-Reduce. In this phase, the Map function calls a user-defined model to handle the <key, value> key, and then generates the middle <key, value> key.

(3) Shuffle: the middle <key, value> key is obtained by Map in shuffle phase, according to the value of key to sort the input data.

(4) Combine: When the map operation outputs its pairs, they are already available in memory. For efficiency reasons, sometimes it makes sense to take advantage of this fact by supplying a combiner class to perform a reduce-type function.

(5) Reduce: The framework calls the application's Reduce function once for each unique key in the sorted order. The Reduce can iterate through the values that are associated with that key and produce zero or more outputs.

(6) Output: The Output Writer writes the output of the Reduce to the stable storage, usually a distributed file system.

As shown in Fig. (**1**), Map-Reduce is efficient in making concise model, good scalability, fault tolerance and parallelism.

2.4. HDFS

HDFS is a storage structure of a distributed computing. It provides input, output data for Map-Reduce parallel computing stage. An HDFS cluster is composed of a number of NameNode and DataNode; these two types of nodes are Master and Worker, respectively. The NameNode is responsible for maintenance tasks of namespace directory and index file and participates in the cluster environment scheduling in a cluster system. DataNode is mainly responsible for the nodes for data storage and task execution, and at the same time, uninterrupted implementation and transmission data report is transmitted through the heart (HeartBeat) mode to NameNode .

2.5. Clustering Genetic Parallel Algorithm based on M-R Model

Clustering centers obtained are optimized by genetic algorithm to avoid the unreasonable clustering analysis in screening the initial cluster center of clustering results and convergence rate of the problem. The main work of algorithm is to calculate the distance of sample to the cluster center and redistribution of the cluster, and the calculation of different clusters is independent. According to the independent characteristics of each group, this paper presents a parallel K-Means algorithm by the computing model of the MapReduce. The design of algorithm mainly includes the clustering center optimization function, Map, Combine, Reduce function. The algorithm process is shown in Fig. (**2**).

3. THE IMPLEMENTATION OF IMPROVED K-MEANS ALGORITHM

3.1. Screening the Optimal Clustering Center

K-Means clustering algorithm has the characteristics of local search, but its clustering convergence speed and its

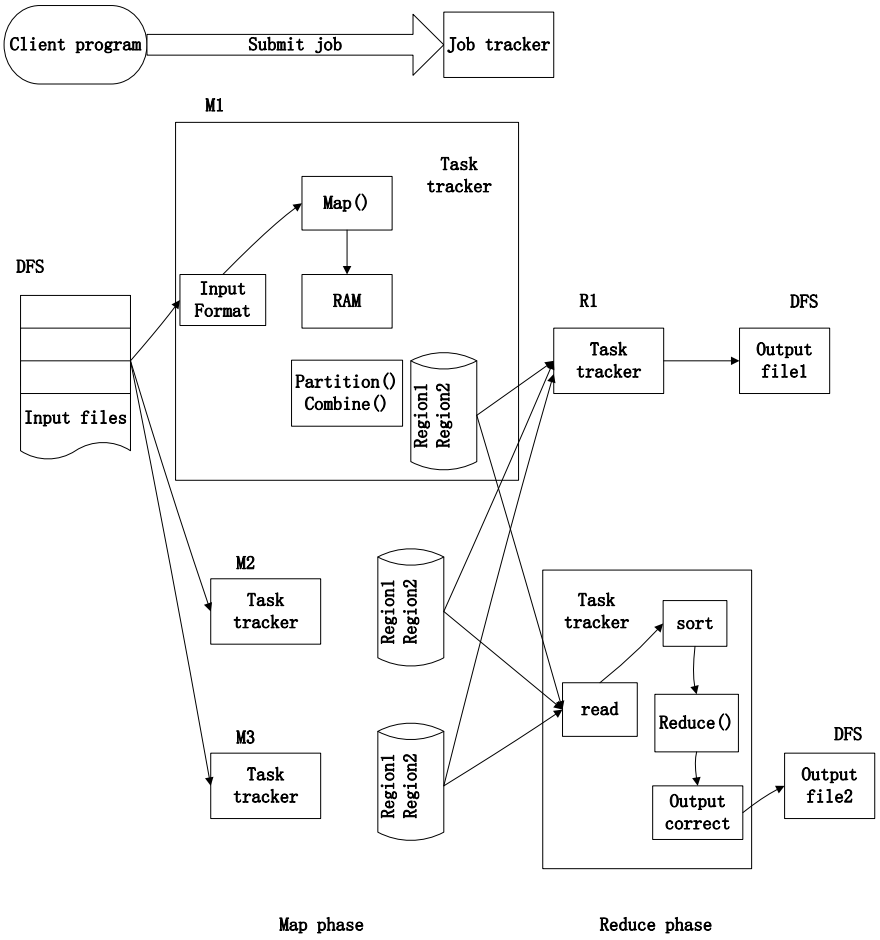

Fig. (1). The computing model of map-reduce.

effective clustering results are affected by the selection of initial clustering center. Once the initial value selection is not good, it is difficult to get ideal results. Genetic algorithm is an adaptive global optimization algorithm. This paper introduces genetic algorithm in K-Means. The influence of the clustering result is no longer affected by the initial clustering center through pretreatment of the clustering center selection. The global optimal cluster centers are found by chromosome coding, fitness function selection and genetic operator of genetic algorithm. The specific steps are as follows.

Step 1: Chromosome code selection. Chromosome code selection directly influences the efficiency of the genetic algorithm and the final result. The binary coded mode is regarded as the solution set of the problem which is mentioned in the literature [11], but the accuracy is not high enough in the process of large-scale numerical optimization. According to the characteristics of large amount of data and complex information, this paper adopts real coding. The code will need the chromosome length as the number of cluster centers, assuming the chromosome length is Li, the coding population for X(k), where k is the first generation of population, then the i-generation population is X(k)i={X(k)1,X(k)2,…,X(k)i}, Ki is the number of cluster centers in the chromosome.

Step 2: Population initialization. Sample K for evolution is randomly selected from the date sample, in order to avoid the premature end of genetic operation; the desired effect of optimal solution cannot be achieved as K cannot choose too small.

Step 3: The appropriate function selection. Sort criteria of candidate hypotheses are defined by the fitness function, and the next generation population criterion is chosen with a certain probability. In this paper, the fitness function is constructed by referencing the K-Means criterion function. The K-Means criterion function and fitness function are as follows:

$$E = \sum_{i=1}^{k} \sum_{x_j \in s_i} (X_j - u_i)^2 \tag{4}$$

$$f = \frac{1}{1+E} \tag{5}$$

E represents the square error of all objects; The Ui represents the average of the Si value, namely, the clustering center; Xj represents the J class sample space; and K is the number of cluster centers. According to clustering criterion F, the more less the E value is, the more excellent is the clustering quality .

Step 4: Genetic operator.

Selection: The individual probability is obtained by individual to individual and group fitness ratio. The choice of individual method is called fitness proportional selection or rotary table selection. Calculation formula of individual is selected as follows:

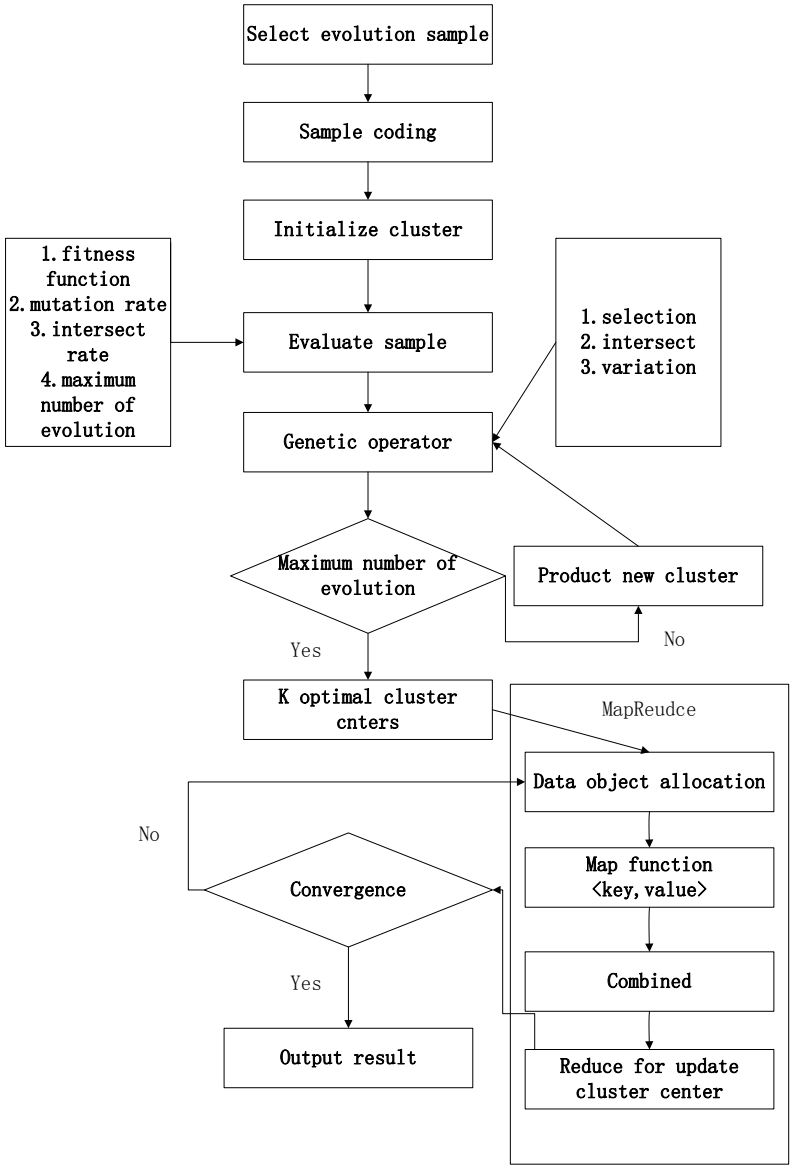

Fig. (2). Flow chart of the algorithm.

$$p(h_i) = \frac{Fitness(h_i)}{\sum_{j=1}^{n} Fitness(h_j)} \tag{6}$$

This method can guarantee the randomness, higher fitness is selected and the smaller is eliminated.

Intersection: According to the real number coding of the log mining, this paper uses uniform crossover method. The method is to randomly select two individuals, taking two intersections in a random way. When the random number is 0, the front part of the individual is cross; when the random number is 1, the middle part of the individual is cross; when the random number is 2, the tail part of the individual is cross.

Variation: According to the characteristics of mass data, each gene selected sample chromosome represents a cluster center. In order to meet the genetic algorithm based on assurance of global search which has also local search performance and the diversity of the group, a mutation operator is introduced. According to the variation of the probability, algorithm selects variant of the individual in the group, then the samples of gene segment have random variation (Here is a selection of uniform probability as the mutation probability). Note that variation range should be in the range of gene segments.

Step 5: Output the optimal initial clustering center and the clustering description information.

Input: The initialization of population N, the probability of crossover and mutation.

Output: The best cluster center, the clustering center description.

// Calculation of each individual fitness, selection probability, the expected probability

for each i∈ range do

CalAllGens(i);

while maxGens do

// Initial population

```
Initial(i);
// Selection of high fitness to enter next generation
selectGens(i);
// Intersection
Crossover();
// Variation
Mutate();
maxGens--;
// Output the optimal clustering center set
printBestResult();
```

3.2. Design of the Map and Reduce functions

The parallel computing task in each MapReduce task is initially a Job. Each Job link can be split into two parts: Map and Reduce stage. The clustering center during the implementation of K-Means is temporarily stored in HDFS; data exchange format uses the TextInputFomat function to regulate. Key represents the clustering center; value represents Euclidean distance of cluster sample to the cluster center.

Input: The clustering center point optimization and clustered sample.

Output: Output clustering is set when the conditions of convergence are satisfied.

Information is mainly described on the map stage and record belongs to the new cluster. The calculation of the shortest distance is carried out between the input sample and the clustering center.

```
1. Class Mapper
2. method Map(LongWritable K,Text V,Context C)
// random sample of input segmentation
3. for each logfiles f ∈ V do
// Traversing the clustering center, gets the short information of the cluster center
4.  for each center i ∈ centers do
5.  if (min more-than distance)
// record minimum
6.  Save distance,pos
7. end for
// Record category and attribute vector set of clustering
8. emitResult(Text(pos), Text(distance)
```

In the Combine stage, the encoded data files have duplicate data, in order to reduce the network flow, while the intermediate results are local mergers. With the same key value of intermediate results, namely clustering center subscripts the same <key, value> key to merge into a group.

```
9. Class Combine
10. Method Combine(LongWritable K,Text V,Context C)
11. EmitLocalResult(Text(pos), Text(distance)
```

At the entrance Reduce phase, data is derived from the Map phase of <key, value> character key queue, where key is the index of cluster center and value is the Euclidean distance of the cluster family. The key of the same sample value is accumulated and calculates the average value, and then the mean value is regarded as the clustering center at the next iteration, and stored in the HDFS.

```
12. Class Reducer
13. Method Reduce(LongWritable K,Iterable V,Context C)
14. sum←0, ave ← 0
15. for each V in Iterable<rs>
16.  sum+=V(value)
17. Ave = sum/count
// Calculate the average value and save it to the initial clustering center set in the next round
18. EmitNextResult(K,ave)
19. Class MRkmeans
// Implementation of the algorithm and output results
20. Job(configration,alg)
```

The algorithm terminates conditions: The last round of the results is compared with the results of this clustering. If the clustering results do not change or are less than the threshold value, the conditions for convergence are satisfied, clustering algorithm is terminated, whereas the results would serve as the next round of the K-Means input.

4. EXPERIMENT AND ANALYSIS OF RESULTS

4.1. Set up the Experimental Environment

By building the Hadoop cloud computing platform to validate the effectiveness of the algorithm, the architecture is shown in the figure. The hardware includes five PC machines, one machine as a master node, loading scheduling and real-time monitoring of task, the remaining four machines as a slave node, loading distributed processing of the task. Each node configuration of The Hadoop platform is shown in Table **1**.

Namenode is regarded as the main control (JobTracker), which is task scheduling and management for distributed clusters; Datanode is regarded as TaskTracker, responsible for performing the task.

4.2. Analysis of Results

According to the different allocations of hardware and software for the two class statuses, this paper mainly studied several experiments. Data source collection is from log files of company server.

Experiment 1: Pseudo distributed cluster mode

Firstly, build a good pseudo distributed platform, run the traditional clustering algorithm (K-Means), a parallel clustering algorithm (CPA) and genetic clustering algorithm (M-R CPGA), to process the size scale of log file as 2MB, 4MB, 6MB, 8MB, 10MB, then compare the execution time, as shown in Fig. (**3**).

Table 1. Configuration of Hadoop platform.

Name	IP	Function	Operation System
hadoop-master	192.168.146.130	namenode datanode	Ubuntu Server 12.04
hadoop-slave1	192.168.146.131	datanode	Ubuntu 12.04
hadoop-slave2	192.168.146.132	datanode	Ubuntu 12.04
hadoop-slave3	192.168.146.133	datanode	Ubuntu 12.04
hadoop-slave4	192.168.146.134	datanode	Ubuntu 11.10

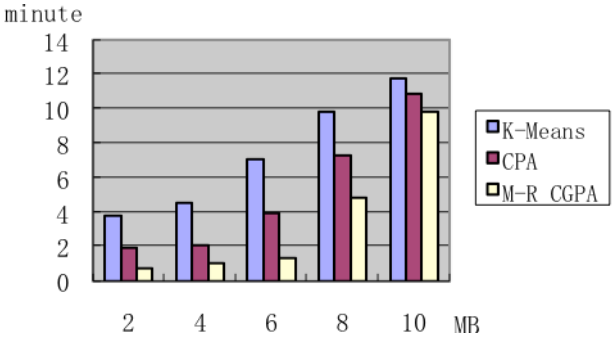

Fig. (3). Execution time of pseudo distributed cluster algorithm.

In Experiment 1, three algorithms are used in different sizes of the log file. The experiments of numerical comparisons show that in a single pseudo distributed case, when a log file is of a very small scale, three kinds of algorithm execution time have an obvious drop; however, data acquisition is very large, because the performance of single configuration cannot reach the fully distributed parallel genetic clustering algorithm. This reflects its disadvantage.

Experiment 2: Distributed cluster mode

In order to validate the algorithm in dealing with more data, thus complete cluster distributed environment needs to be set up in which four machines work as the task processing node while a machine as the task scheduling node. Due to the hardware task, call node configuration is higher but also works as a task node, as shown in Fig. (**3**).

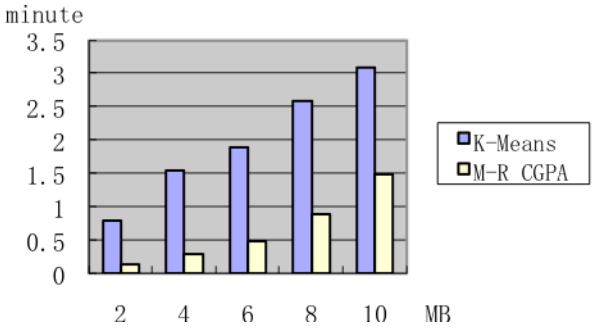

Fig. (4). Execution time of distributed cluster algorithm.

In Experiment 2, the parallel clustering algorithm (CPA) and genetic clustering algorithm (M-R CGPA) were compared. The experimental results show that in the case of M-R cluster distributed CGPA, algorithm shows good speedup. As the data size increases, computing clusters distributed advantages become more prominent, and the scalability of Hadoop platform ensures high availability program algorithm as shown in Fig. (**4**).

CONCLUSION

This paper combined the global optimization of genetic algorithm and K-Means algorithm of local searching characteristic and proposed a genetic clustering based on the calculation model of M-R parallel algorithm. Through the deployment of Hadoop cluster environment, the algorithm ran in HDFS and the massive log files were processed. The experimental results show that with the cluster nodes and the amount of data increase, the efficiency of the algorithm is higher to achieve effect of mining on massive log, laying the foundation for future study of combination in the massive data off-line calculation and multidimensional data mining.

CONFLICT OF INTEREST

The authors confirm that this article content has no conflict of interest.

ACKNOWLEDGEMENTS

This work is supported by Guangxi key Laboratory of Trusted Software (No: kx201317), by the Postgraduate's Innovation Project of Guilin University of Electronic Technology under (No: GDYCSZ201470), by the 2014 Guangxi University of Science and Technology Research Projects (NO: LX2014149), by the Nature Science Foundation of Guangxi (No: 2013GXNSFAA019350).

REFERENCES

[1] H. Yu, and D. Wang, "Mass log data processing and mining based on Hadoop and cloud Computing", In: *Computer Science & Education* (ICCSE), 2012.

[2] Y. Liu, N. Cao, W. Pan, and G. Qiao, "System anomaly detection in distributed systems through MapReduce-Based log analysis", *Advanced Computer Theory and Engineering* (ICACTE), vol. 6, pp. 410-413, 2010.

[3] S. Bandyopadhyay, "Genetic algorithms for clustering and fuzzy clustering", *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 3, p. 285, 2012.

[4] X. Qin, and H. Wang, "Big Data Analysis:Competition and symbiosis of RDBMS and MapReduce", *Journal of Software*, vol. 23, pp. 32-45, 2012.

[5] X. Meng, and X. Ci, "Big data management: conception, technology and challenge", *Research and Development of Computer*, vol. 50, pp. 146-169, 2013.

[6] S. Wang, H. Wang, and X. Qin, "Big data structure: challenge, present situation and prospect", *Journal of Computer*, vol. 10, pp. 1741-1752, 2011.

[7] Q. Song, and J. Shen, "Efficient mining algorithm of web log", *Research and Development of Computer*, vol. 3, pp. 328-333, 2001.

[8] C. Zhang, and H. Ying, "Uniform block Crossover Genetic Algorithm", *Technology and Application of Automation*, vol. 24, pp. 17-23, 2005.

[9] Y. Wu, "Discussion on custering analysis method", *Science of Computer*, vol. 39, pp. 325-327, 2012.

[10] Y. Ma, and W. Yun, "Research Progress on genetic algorithm", *The Research and Application of Computer*, vol. 29, pp. 1201-1210, 2012.

[11] R. C Taylor, "An overview of the Hadoop/MapReduce/HBase framework and its current applications in bioinformatics", In: *11th Annual Bioinformatics Open Source Conference* (BOSC), Boston, MA, USA, 2010.

Received: September 16, 2014 Revised: December 23, 2014 Accepted: December 31, 2014