Journal of Computers Vol. 29 No. 1, 2018, pp. 47-62
doi:10.3966/199115992018012901005

Research on Differential Privacy Preserving Clustering Algorithm Based on Spark Platform

Qianqian Meng[1*], Lijuan Zhou[2]

[1] Department of Information Engineering College, Capital Normal University, Beijing, China
lnbxxmqq@163.com

[2] Department of Information Engineering College, Capital Normal University, Beijing, China
zlj87@tom.com

Received 16 September 2016; Revised 12 March 2017; Accepted 26 March 2017

Abstract. Differential privacy is a kind of privacy protection model based on data distortion proposed by Dwork. As the model does not need to assume the prior knowledge of the attacker, it has been a research hot spot in the field of privacy protection. Aimed at the problem that the traditional differential privacy K-means algorithm is more sensitive to the selection of the initial center points, which reduces the usability of clustering results, an improved differential privacy preserving clustering algorithm (DEDP K-means) is proposed by introducing adaptive opposition-based learning technique and differential evolution algorithm. At the same time, the improved algorithm is parallelized based on the Spark platform. It was also demonstrated that the improved algorithm can optimize the selection of the initial centers, improve the usability of clustering results and have a good speedup when dealing with massive data by parallel experiments.

Keywords: differential evolution, differential privacy, k-means, opposition-based learning, spark

1 Introduction

With the rapid development of Internet technology, the whole society was forced into the era of big data. As we all know, these huge amounts of data implies great value. Enterprises analyze users' data through data mining technology. The results of the analysis for future decision-making can bring great benefits to the enterprise. In recent years, the concepts of big data mining and analysis, big data precision marketing and advertising precision delivery rise rapidly. However, the objects of data mining often contain some privacy information, such as the user's bank card information, medical information, home address and other information. Once the privacy data is excavated by malicious attackers, it will cause irreparable damage to the user. Nowadays, the frequent occurrence of privacy disclosures has aroused concerns of users, more and more users refuse to provide real data, which brings new challenges to data mining technology. Privacy preserving technology based on data mining is a hot research topic. How to protect the privacy of users and improve the usability of data mining is an urgent problem to be solved.

In view of the above problems, privacy protection technology arises at the historic moment. Privacy preserving data mining [1], which was first proposed by Agrawal et al., is widely used in published data privacy protection and privacy protection in data mining. At present privacy protection technology is divided into three categories: data distortion, data encryption and limit release [2]. The purpose of data distortion is to disturb or randomize the data by adding noise to the original data set. But at the expense of the accuracy and authenticity of the data. Data encryption is used to hide sensitive data of the objects. This method can guarantee the accuracy of data, but requires a large amount of memory overhead. Restricted release is based on the specific circumstances of the data, the privacy of information is encrypted or deleted and then released, the method makes the published data have certain information

* Corresponding Author

which is missing. Based on the above techniques, scholars have put forward a variety of privacy protection models. Among them, K-anonymity model and L-diversity model are widely used. The K-anonymity model was put forward by Sweeney et al. [3-4], which divides *k* records that can not distinguish with each other into an equivalence class. To ensure that any records with other *k-1* records can not be distinguished. The greater the value of *k*, the less risk of privacy disclosure, but the more information is lost. K-anonymity algorithm is prone to information leakage under homogeneity attacks and background knowledge attacks. The L-diversity model proposed by Machanavajjhala [5] avoids the problem that the sensitive attribute value is single in the same equivalence class, so the risk of leakage is less than *1/L*, but the model is susceptible to similarity attack. However, the model is vulnerable to similarity attack. K-anonymity and L-diversity privacy models can reduce the risk of privacy disclosure to some extent. However, both models need to assume the background knowledge of attackers and can not quantify the level of privacy protection through the parameters.

Dwork proposed a new privacy protection model in 2006 - Differential Privacy [6], which protects the sensitive data by random scrambling of data. The differential privacy protection model defines an extremely strict attack model that quantifies the level of privacy protection through the privacy protection budget ε. As a new research hot spot, differential privacy protection has very important value both in theoretical research and in practical application. For the studies of differential privacy in clustering analysis are less currently, Blum et al. [7] first proposed that achieve privacy protection in the K-means clustering process by adding appropriate noise - Differential Privacy K-means Algorithm(DP K-means), and implemented on the SuLQ platform. The paper summarized the steps of obtaining ε - differential privacy K-means algorithm, but did not show how to set the privacy protection budget ε, which reduces the usability of the results. On the basis of Blum et al., Dwork [8] analyzed the differential privacy K-means algorithm in detail and gave two ways to set the privacy protection budget ε. Since the DP K-means algorithm was put forward, many scholars have studied the algorithm. In the paper [9], since the number of clusters is difficult to estimate, then it combined K-means algorithm with Canopy algorithm, the output is used as the input of the K-means algorithm, which solves the problem of determining of the value of *k* and avoids the effect of isolated points on clustering results. For the DP K-means algorithm depends on the initial clustering centers, the central estimation algorithm based on mean density is used to estimate the initial cluster centers in the paper [10], the paper [11-12] selects the *k* points that are farthest away from each other in high density areas as the initial clustering centers.

Today, with the outbreak of data, massive and high dimensional data have brought new challenges to the clustering algorithm, and single machine processing has been unable to meet the demand for speed. Therefore, it is necessary to parallelize the algorithm running on a large data processing platform. Mao Paper [13] put forward a kind of K-means algorithm based on MapReduce, which runs on Hadoop platform in parallel. Using HDFS distributed data storage and MapReduce distributed computing framework to deal with K-means algorithm. The Hadoop platform can make full use of the clusters to process the program in parallel, and calculate the clustering results efficiently [14-16]. However, MapReduce [17] is a disk-based batching framework that requires repeated disk storage, reads, and other operations during each computation. As a result, a MapReduce operation consumes a significant amount of time during the processing of dealing with massive data.

In this paper, we propose an improved privacy preserving K-means algorithm - DEDP K-means algorithm, which is based on the satisfaction of ε - differential privacy protection, the adaptive Opposition-based Learning and differential evolution algorithm, to solve the problem of poor clustering results in usability. The improved algorithm optimizes the selection of initial center points, reduces the number of iterations and improves the usability of clustering results. At the same time, the improved algorithm is parallelized in the framework of Spark, which makes the algorithm have better speedup when dealing with massive data and high-dimensional data.

2 Related Works

The main research problems of the paper are divided into the following aspects: In this paper, we focus on the widely used differential privacy protection model. There are two aspects of research on differential privacy protection technology. First, the privacy protection of data release and the other is the privacy protection in data mining. This paper mainly studies the privacy protection in data mining. Clustering is one of the important data analysis techniques in data mining. In this paper, for the problem of privacy

disclosure caused by the classical clustering algorithm K-means in calculating the distance between the data point and the center point, the K-means clustering algorithm based on differential privacy protection is described in detail. The traditional DP K-means clustering algorithm relies on the initial center point selection, which leads to the poor availability of clustering results. In this paper, a series of optimization strategies are proposed, through the experimental results to prove its feasibility. The improved clustering algorithm is parallelized on the Spark platform, which makes the algorithm suitable for massive data.

3 Differential Privacy Protection

3.1 The Definition of Differential Privacy

Differential privacy is a privacy preserving model proposed by Dwork in 2006, which is based on data distortion technology. Differential privacy preserving model overcomes two major shortcomings of traditional models:

(1) A strict attack model is defined. Assuming that the attacker possesses the greatest background knowledge, even if the attacker masters all the records except the target record, the target record will not be leaked, which can deal with malicious analysis under arbitrary background knowledge[18].

(2) A strict definition and quantitative evaluation method of privacy protection levels are given.

In the case of differential privacy protection, the addition or deletion of a record in the data set will not affect the output of the query. Here are some basic definitions of differential privacy protection.

Definition 1: Suppose that D and D' have the same attribute structure, the symmetric difference between them is written as $D \triangle D'$. $|D \triangle D'|$ represents the number of discrepant records. If $|D \triangle D'| = 1$, then D and D' are called adjacent data sets.

Theorem 1 Differential Privacy [6]: A stochastic algorithm K is given, P_K represents the set of all possible outputs of the random function K, and $Pr[Es]$ represents the disclosure risk of event Es. For any adjacent data sets $D1$, $D2$ and any subset Sk of the P_K, if the algorithm satisfies the following inequality:

$$\Pr[K(D1 \in Sk)] \leq \exp(\varepsilon) * \Pr[K(D2) \in Sk] \ . \quad \textbf{(1)}$$

Then it is said that the algorithm provides ε - differential privacy protection. The parameter ε is called the privacy protection budget. It can be seen from the above equation, the lower the ε value is, the smaller query results on the adjacent data sets $D1$ and $D2$, the higher level of privacy protection provided by the random function K is. Fig. 1 depicts the disclosure risk curve for the adjacent data sets $D1$ and $D2$ which satisfy ε -differential privacy protection.

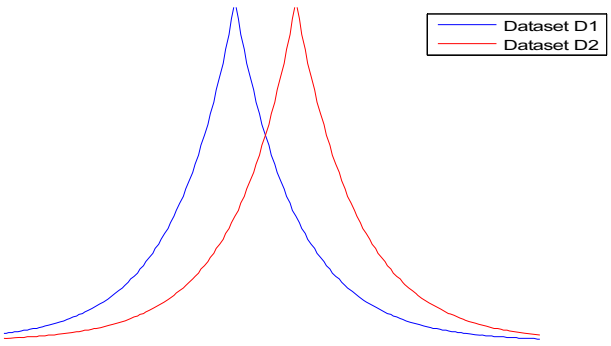

Fig. 1. The disclosure risk curve of ε - differential privacy protection

3.2 Differential Privacy Implementation Mechanism

The key point of differential privacy technology is the noise mechanism. Adding random noise to the query results can prevent the attacker from deducing the target information from the background knowledge, so as to achieve the purpose of protecting the sensitive information. At present, the

commonly used noise mechanisms are the Laplace mechanism and the Index mechanism, Laplace mechanism is adopted in this paper.

The Laplace mechanism achieves ε - differential privacy protection by adding noise that obeys the Laplace distribution to the query results.

Definition 2: If the probability density function of the random variable x is distributed as:

$$p(x) = \frac{1}{2 * b} \exp(-\frac{|x - u|}{b}). \tag{2}$$

x is said to obey the Laplace distribution, denoted as x~Lap (u, b), where u is the location parameter, b is the scale parameter.

Definition 3 Query Sensitivity: The sensitivity of the query function f: D->R_k is defined as:

$$\Delta f = \max_{D1, D2} |f(D1) - f(D2)| . \tag{3}$$

As can be seen from the above, the sensitivity is the inherent property of the query function, only related to the function itself, and independent of the size of the data set.

Theorem 2 Laplace mechanism: For any query function f, f (D) is the query results on data set D, and ε -differential privacy protection is realized by adding random noise Y to f (D), where Y~Lap($\Delta f / \varepsilon$) . The final query response of function f is:

$$f(D) + Lap(\Delta f / \varepsilon) . \tag{4}$$

The noise function is $Lap(\Delta f / \varepsilon) = \exp(-\frac{|x| * \varepsilon}{\Delta f})$ and the probability density function is $Lap(\Delta f / \varepsilon) = \frac{\varepsilon}{2 * \Delta f} \exp(-\frac{|x| * \varepsilon}{\Delta f})$. It can be seen that the amount of noise is inversely proportional to the privacy protection budget ε, which is proportional to the query sensitivity Δf. As in the same query function, Δf is a certain value. It can be seen in Fig. 2 that the smaller the ε is, the greater the noise is, and the higher level of privacy protection is.

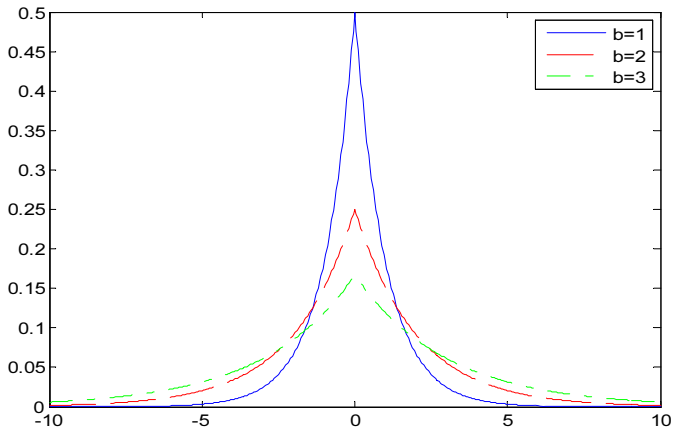

Fig. 2. The Laplace probability density function under different ε

4 Differential Privacy K-means Algorithm

4.1 K-means Algorithm

K-means algorithm is a classical clustering algorithm based on division proposed by Mac-Queen [19].The basic idea of this algorithm is that for a data set containing N objects, we finally find k cluster centers, and divide the remaining objects into the nearest cluster, so that the distance between the data points and the cluster center is minimum in each cluster.

The basic steps of algorithm are as follows:

Input: cluster numbers k, the initial centers and data set $D=\{x_1, x_2, ..., x_n\}$

Output: k cluster members

Step 1: The initial cluster centers are randomly selected from the data set.

Step 2: Compute Euclidean distance (formula (5)) between remaining objects and each cluster center. According to the principle of proximity, the object is divided into the appropriate cluster.

$$d(x_i, x_j) = \sqrt{(x_{i1} - x_{j1})^2 + ... + (x_{in} - x_{jn})^2} \tag{5}$$

Step 3: According to each cluster's objects and formula (6), the positions of the k centroids are recalculated, where ω_i is the *i-th* cluster, m represents the number of data belonging to the ω_i cluster.

$$z_i = \frac{1}{m} \sum_{x \in w_i} x \tag{6}$$

Step 4: Calculate the square error according to the formula (7).

$$E = \sum_{i=1}^{k} \sum_{p \in wi} | p - m_i |^2 \tag{7}$$

Step 5: If the function converges, the algorithm is terminated. Otherwise, return to step 2.

The problem of privacy disclosure in K-means algorithm is mainly concentrated in step 2. Assuming that the attacker obtains the distance between a d-dimensional data item and k cluster center points in the m iterations, then the number of $m*k$ equations are obtained. If $m*k>d$, the attacker can easily calculate the specific value of each attribute of the data item, leading to privacy disclosure.

4.2 The Traditional ε-Differential Privacy K-means Algorithm-DP K-means

For the traditional K-means algorithm will produce privacy leaks in the calculation of the distance between the data items and the cluster center. Blum et al. proposed a differential privacy K-means algorithm, Dwork improved the algorithm and gave two ways of setting ε value.

The specific steps of the ε- differential privacy protection K-means algorithm are as follows:

Step 1: Input n data items of d-dimensional space $D = \{x_1, x_2, ..., x_n\}$, from which the k initial centers u_1, u_2, ..., u_k are randomly selected. Return the k points u_1', u_2', ..., u_k' that added noise in the d-dimensional space $[0,1]^d$ as the new initial center points.

Step 2: The remaining data items are divided into the nearest cluster center u_i', and the data set is divided into k clusters D_1, D_2, ..., D_k.

Step 3: For $1<= i<= k$, calculate the sum of each data item in D_i $sum = \sum_{x \in D_i} x$, and the number of data items in D_i $num = | D_i |$. Add the noise that follows the Laplace distribution to get sum' and num'. Update cluster center $u_i'' = \frac{sum'}{num'}$.

Step 4: Calculate the squared error E according to formula (8).

$$E = \sum_{i=1}^{k} \sum_{p \in Di} | p - D_i |^2 \ . \tag{8}$$

Step 5: If the function converges, the algorithm is terminated. Otherwise, return to step 2.

The algorithm flowchart is shown as in Fig. 3.

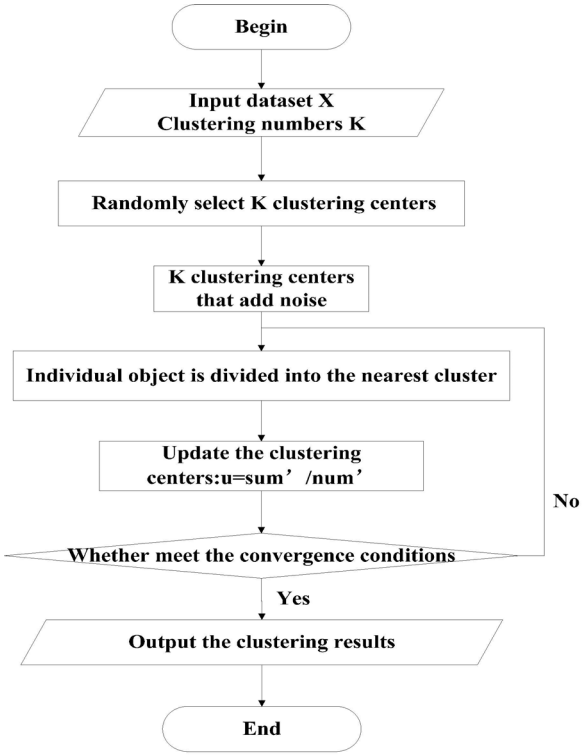

Fig. 3. The algorithm flowchart of DP K-means

5 Improved Algorithm

The DP K-means algorithm solves the problem of privacy disclosure in the clustering process, so that the attacker can not deduce the user’s privacy information through the acquired knowledge. However, a large number of experiments show that DP K-means algorithm has the problem of low accuracy, on the one hand, the initial center points that randomly generated will decrease the convergence rate of the algorithm, on the other hand, if the random center point is added with noise, the deviation between the center points and the original center points is increasing, and the availability of the clustering result is reduced. In this paper, an improved strategy is proposed to optimize the selection of the initial center points by introducing the differential evolution algorithm and the adaptive reverse learning technique to increase the usability of the differential privacy clustering results.

5.1 Differential Evolution Algorithm

Differential evolution algorithm (DE) is a kind of heuristic search algorithm based on population, which is proposed by Storn et al. [20]. The algorithm is based on the difference between the population, using the random search method, through multiple iterations, the better fitness of the individual will be remained. The basic idea of the DE algorithm is to carry out the differential mutation operation and discrete crossover operation to the initial population, and to obtain the newly generated experimental population. Finally DE adopts the greedy selection strategy to update the population. Then the better individual is picked out as the parent generation to the next iteration by comparing the fitness value of generated experimental population and initial population. The evolution of the DE algorithm flow chart is shown in Fig. 4 below.

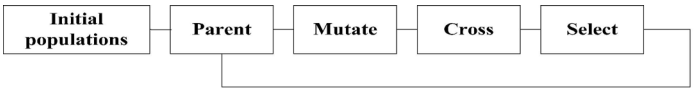

Fig. 4. DE algorithm evolutionary process

5.2 Adaptive Opposition-based Learning Technique

Tizhoosh [21] firstly proposed the concept of opposition-based learning in 2005, this technology has been proposed as a new scheme in the field of machine intelligence in recent years [22-24]. General swarm intelligence algorithms randomly choose initial value [25], then approach to the optimal solution with each generation optimization. Initial value has much effect on the algorithm, if the initial value is close to the global optimal solution, then algorithm converges quickly, whereas algorithm will be more time- consuming and easy to fall into local optimum. Opposition-based learning provides a new idea in the problems of optimizing the initial value. In the process of search, select the optimum solutions into the next generation by simultaneously evaluate the current population and the opposite one, thus widening the search area, improving the diversity of the population and accelerating the convergence speed.

Algorithm defined:

(1) Opposition-based learning point: Suppose x is a real number which is between a and b, the reverse point $\tilde{x}$ of x is: $\tilde{x} = a+b-x$.

(2) Opposition-based learning vector: Suppose $P=(x_1, x_2, ..., x_n)$ is a space point in n dimensional vector space, in which $x_1, x_2, ..., x_n \in R$ and $x_i \in [a_i, b_i]$. Then the reverse vector of P is defined as $\tilde{P} = (\tilde{x}_1, \tilde{x}_2, \tilde{x}_3, ..., \tilde{x}_n)$, in which $\tilde{x}_i = a_i + b_i - x_i$.

(3) Opposition-based learning mechanism: Suppose $f(x)$ is the fitness function, calculating $f(x)$ and $f(\tilde{x})$ in each iteration of the algorithm, if $f(\tilde{x}) > f(x)$, then $\tilde{x}$ replaces x in the next iteration.

5.3 Adaptive Crossover Operator

Differential evolution algorithm uses crossover operation to maintain the diversity of the population, but with the evolution of the population, the differences between individuals are getting smaller and smaller, and the diversity of the population has decreased rapidly, and the phenomenon of premature has come out. The crossover operator CR controls the contribution of the initial population and the mutated population to the experimental population. The larger CR is, the more contribution of the mutated individual has, which is beneficial to the local search and speed up convergence rate. The smaller the CR is, the greater the contribution of the initial population to the experimental population is, which is conducive to maintain the diversity of the population and the global search [26]. Thus, it is not advisable for the traditional differential evolution algorithm to set the crossover CR as a fixed value. In order to coordinate the global search ability and local search ability of the algorithm, this paper constructs a dynamic adaptive crossover operator:

$$CR = CR_{\min} + (CR_{\max} - CR_{\min}) * e^{-20*(1-g/(G\max+1))}. \quad (9)$$

where CR_{min}= 0.25, CR_{max} = 0.95, G_{max} = 50, and g represents the current evolutionary algebra. It can be seen from Fig. 5 that CR mainly goes through three stages in the process of evolution: firstly, CR keeps a small value, mainly for global search. Then CR gradually increases, reducing the search area. Finally CR is stable at a large value stage and perform a local search. In this paper, the dynamic adaptive crossover operator can make the global search ability and the local search ability in a good balance, improve the early fall of the algorithm into the local optimal solution, and reduce the dependence of the algorithm on the crossover operator. The algorithm improves the early maturation of the local optimal solution and reduces the dependence of the algorithm on the parameters of the crossover operator.

5.4 Improved Differential Privacy K-means algorithm—DEDP K-means

For the problem that the clustering result of DP K-means algorithm depends on the initial center points, this paper proposes a strategy to optimize the initial clustering center by using DE algorithm. The DE algorithm has the advantages of simple, high efficiency and good robustness, but with the increase of the evolution algebra, the differences between populations are gradually reduced, the convergence rate of the algorithm is slowed down, and the premature phenomenon is easy to occur. In order to solve this problem, this paper introduces opposition-based learning technology to improve the search ability of the algorithm by evaluating the current solution and the reverse solution to avoid the premature phenomenon.

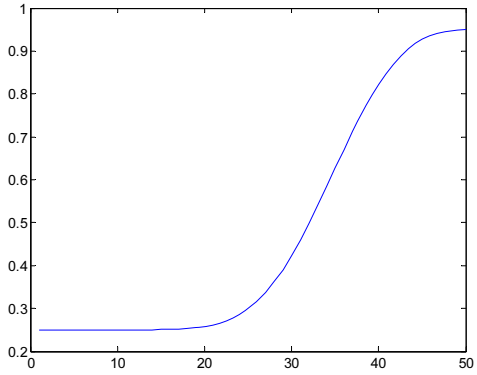

Fig. 5. The change curve of CR

In order to maintain the population diversity of DE algorithm, an improved algorithm based on adaptive opposition-based learning is proposed in this paper. Suppose the current individuals of the population are $X=(x_1, x_2, ..., x_n)$, reverse solution is constituted according to (10), in which $x_i \in [a_i, b_i]$ and $k \in [0, 1]$. Generalized coefficient k is used to generate reverse individuals in different area. The dynamic boundary replaces the traditional fixed boundary in the process of search that is more beneficial to improve the local exploitation ability of the algorithm and make the algorithm have a greater chance of converge to global optimal solution.

$$\widetilde{X}_i = k * (a_i + b_i) - X_i \quad . \tag{10}$$

Improved algorithm detailed steps are as follows:

(1) Initialize parameters: k is the number of clusters, Np is the population size, CR is the crossover probability, F is the scaling factor, $Tmax$ is the number of differential evolution generations, Tk is the number of K-means evolution generations, ε is the privacy budget.

(2) Initialize population: k samples are selected randomly from the data set to be centralized as a group of initial cluster centers. Repeat Np times, then Np groups of clustering centers are obtained. Real number coding is used in the algorithm and the initial evolution generation is zero. Specific encoding is as follows:

$$X_j(0) = (X_{j,1}, X_{j,2}, ..., X_{j,k}) . \tag{11}$$

where $j = 1, 2, ..., Np$, $X_{j,i}$ represents the j-th individual's i-th cluster center.

(3) Mutation: The paper chooses "DE/rand/2/bin" mutation strategy (formula (12)). Formula (12) is as follows, in which $a \neq b \neq c \neq d \neq j \in [1, Np]$. F is the scaling factor. Mutated individual is represented as $Vj(g+1)$.

$$V_{j,i}(g+1) = X_{e,i} + F[(X_{a,i} - X_{b,i}) + (X_{c,i} - X_{d,i})] \quad . \tag{12}$$

(4) Crossover：Experimental individual $Mj1(g+1)$ is obtained through crossover operation on the current individual $Xj(g)$ and mutated individual $Vj(g+1)$ according to fromula (13). $\beta \in [0,1]$ is a random decimal which is generated when comparing each gene. $\gamma \in [1, D]$ is a random integer. CR is the crossover probability.

$$M_{j1,i}(g+1) = \begin{cases} V_{j,i}(g+1) \ if \ \ \beta \le C_R \ or \ \ i = \gamma \\ X_{j,i}(g) \ \ else \end{cases} \quad . \tag{13}$$

(5) Opposition solution: Experimental individual $Mj2(g+1)$ is obtained through adaptive opposition-based learning on the experimental individual $Mj1(g+1)$ according to (10).

(6) Selection: Calculates the objective function value of the current population $X_j(g)$, the experimental population $M_{j1}(g+1)$ and the experimental population $M_{j2}(g+1)$ according to the squared difference criterion function. Select the optimum population into the next generation according to (14) .

$$X_j(g+1)=\begin{cases} X_j(g) & f(X_j(g)) < f(M_{j1}(g+1)) \,||\, f(X_j(g)) < f(M_{j2}(g+1)) \\ M_{j1}(g+1) & f(X_j(g)) > f(M_{j1}(g+1)) \,||\, f(M_{j2}(g+1)) > f(M_{j1}(g+1)) \\ M_{j2}(g+1) & else \end{cases} . \quad (14)$$

(7) Constantly repeat (3) ~ (6) steps until the function converges or the number of iterations is up to *Tmax*. Finally, output a set of optimal values as the initial cluster centers.

(8) The optimized clustering centers are used as inputs of DP K-means algorithm.

Algorithm flow chart is shown in Fig. 6.

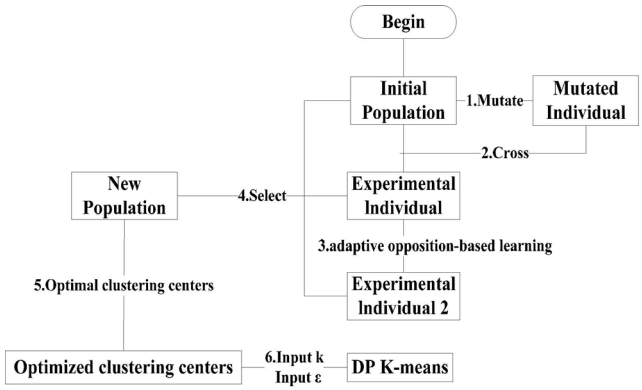

Fig. 6. DEDP K-means flow chart

5.5 Algorithm Framework

The parameters involved in the algorithm are shown in the following Table 1.

Table 1. Description of the symbols in the algorithm

The symbol	Description
$D=\{x_1, x_2, ..., x_n\}$	Initial data set
k	The number of clusters
T_{max}	The maximum evolutionary algebra
XG	Initial population
XG_next_1	Mutated population
XG_next_2	Crossed population
XG_next_3	Elite reverse population
$XG\_next=\{u_1,u_2,...,u_k\}$	Optimized cluster centers
ε	Privacy protection budget

```
Begin
t=1;
Initialize each parameter in the algorithm;
Normalize attribute values in data set D;
Initialize a population XG as inital clustering center;
while (t<=Tmax)
{
Randomly generated three individuals Xa.Xb,Xc;
XG_next_1=Xa+F*(Xb-Xc);
for j=1:D
{
         if (rand>CR & randx(1)~=j)
          XG_next_2=XG;
         else
          XG_next_2=XG_next_1;
         end
}
XG_next_3=k*(max(XG_next_2)+min(XG_next_2))-XG_next_2;
 if fit(XG_next_2) < fit(XG) && fit(XG_next_2)<fit(XG_next_3)
```

```
          XG_next=XG_next_2;
 else
        if fit(XG_next_3) < fit(XG) && fit(XG_next_3)<fit(XG_next_2)
         XG_next= XG_next_3;
     else
          XG_next= XG;
     end
 end
Calculate the fitness value of each individual in XG_next;
best_vector=min(f(XG_next));
t=t+1;
}
Put the optimize clustering center into the traditional DP K-means
algorithm.
End
```

6 Parallel Implement Of Algorithm

6.1 Distributed Computing Platform-Spark

Spark is a memory-based large data distributed processing framework proposed by UC Berkeley AMP Lab. Spark implements distributed computing based on MapReduce [27]. Spark has the advantage of Hadoop MapReduce [28], except that the intermediate output is stored in memory, eliminating the need to read and write HDFS. Therefore, Spark can run better in data mining and machine learning algorithm which need iterations. Based on the above conditions, this paper selects the Spark distributed data processing tools with the HDFS distributed file system to realize the parallelization of improved algorithm.

The main core idea of Spark is the Resilient Distributed Dataset (RDD), which is an abstract concept of distributed, allowing developers to perform memory-based computing on large-scale clusters. It is a collection of read-only partitions that can only be generated by reading HDFS or other distributed file systems or converting by other RDD. Spark provides a variety of operations on the data set RDD, which can be divided into Transformation and Action. The conversion operation is inert, that is to say that a RDD to another new RDD conversion operation will not be executed immediately, and is really triggered in the event of Action operation (Fig. 7).

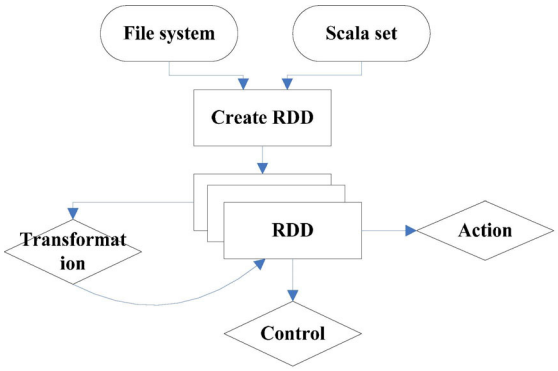

Fig. 7. The operations of RDD

6.2 Parallel Design of Improved DP K-means Algorithm Based on Spark

The K-means algorithm is divided into two iterations: Calculate the distance between each data object and each cluster center in the data set, and its computational complexity is O ($n*k*d$) = n (the number of data) * k (the number of clusters) * d (the data dimension) . The process of updating the cluster center points according to the divided clusters, and the computational complexity is O ($n*d$) .

In order to solve the above problems, the DP K-means algorithm is parallelized. The main steps of the algorithm are as follows (Fig. 8):

(1) Read the data set to be divided from HDFS, assign it to the RDD data set under the Spark framework, and encode the improved initial clustering centers.
(2) The Map operation is performed. Calculate the distance of the data object to each center, and divide it into the nearest class, record the clustering of each data point, constitute <key, value> pairs.
(3) The Join operation is performed. Aggregate data points belonging to the same class, and calculate the sum of the data points and the number of data points in each class.
(4) The ReduceByKey operation is performed. Add Laplace noise to the variable *sum* and *num* and then update the cluster centers.
(5) Determine whether the algorithm converges, if the convergence is over; otherwise, the new RDD dataset is looped through steps 2 through 5 and the resulting intermediate results are cached in Cache.

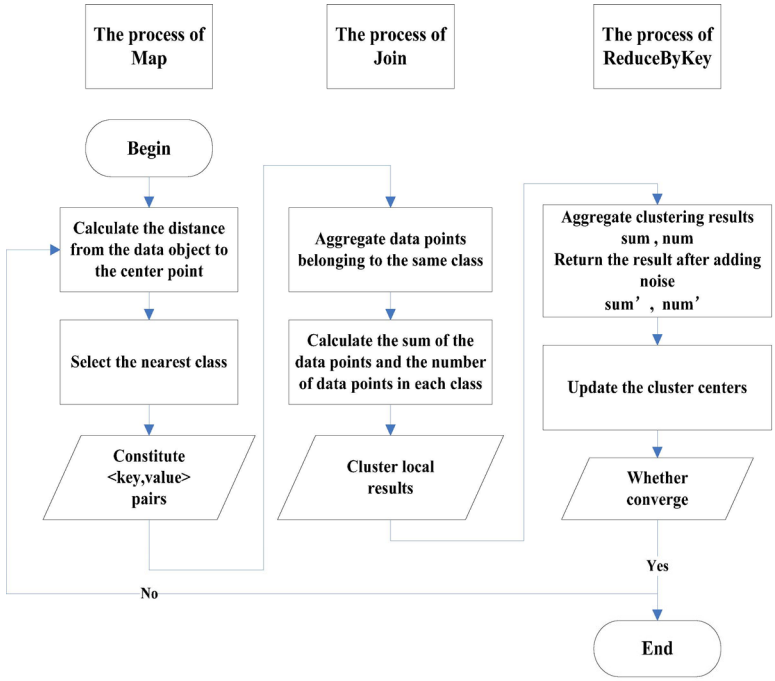

Fig. 8. Parallel flow chart of DEDP K-means

7 Simulation and Analysis-Comparison of Cluster Accuracy

Compare the accuracy of clustering results of the traditional DP K-means algorithm with the DEDP K-means algorithm proposed in this paper. Algorithm is based on Matlab platform, the experimental environment is Windows 7, memory 4GB. The data in the experiment is from UCI Knowledge Discovery Archive database, and the data set properties are shown in Table 2.

Table 2. The attributes of the data set

Name of Data Set	Number of Attributes	Number of Clusters	Number of Instances
Wine	13	3	179 (59,71,48)
Magic Gamma Telescope	11	2	19020 (12332,6688)

7.1 Clustering Evaluation Index F-measure

F-measure [29] is commonly used to measure clustering availability, the main parameters of the F-measure are the accuracy P (Precision) and the recall rate R (Recall). It can be seen from formula (14) and formula (15), two indicators are mutual restriction in large-scale data sets. F-measure is a comprehensive evaluation index, when the F value is higher, the results of the two clustering methods show a higher similarity.

Two methods C and D for clustering a data set, the clustering results were $C_1, C_2, \ldots, C_K, D_1, D_2, \ldots, D_K$. The P and R of cluster C_i and D_j are defined as follows:

$$R = \Pr ecision(C_i, D_j) = \frac{N_{ij}}{|D_i|}. \tag{15}$$

$$R = \mathrm{Re}\, call(C_i, D_j) = \frac{N_{ij}}{C_i}. \tag{16}$$

where $1<= i<=k$, $1<=j<=k$, $|T|$ represents the number of objects in the data set T and N_{ij} represents the number of C_i objects contained in D_j. From the above results, F-measure value of C_i and D_j can be calculated, the formula is as follows:

$$F(i) = \frac{2 * P * R}{P + R}. \tag{17}$$

For a data set, the F-measure value of the entire cluster results is:

$$F = \frac{\sum_i [|C_i| * F(i)]}{\sum_i |C_i|}. \tag{18}$$

7.2 Experimental Results

During the experiment, two data sets were normalized at first. The ε - differential privacy clustering and DEDP K-means clustering were carried out for the two data sets respectively. The change of F-measure was observed with the value of ε from 0.05 to 6. Parameter settings during the experiment: the dimension of the problem to be solved is D, population size is $5*D$, mutation rate F is 0.5, the minimum of crossover probability is 0.25, the maximum of crossover probability is 0.95, the maximum evolution generation $Tmax$ is 10000. In order to reduce experimental error, respectively perform 10 times on DP K-means and DEDP K-means algorithms, take the average value as the final evaluation reference.

Table 3 records the results in the three algorithms on Glass data set.

Table 3. The clustering result of Wine data set

	K-means	DP K-means	DEDP K-means
The minimum distance within class	16555.67	16555.70	16545.31
The maximum distance within class	18437.00	18523.12	17603.00
The average distance within class	18103.71	18284.62	16615.84

Table 4 records the results in the three algorithms on MAGIC data set.

Table 4. The clustering result of MAGIC data set

	K-means	DP K-means	DEDP K-means
The minimum distance within class	1.6697e+06	1.7032e+06	1.6912e+06
The maximum distance within class	1.9184e+06	1.9978e+06	1.8476e+06
The average distance within class	1.8054e+06	1.9015e+06	1.7910e+06

It can be seen from Table 3 and Table 4 that the traditional DP K-means algorithm reduces the accuracy of clustering results in the clustering process. The improved DEDP K-means algorithm proposed in this paper, improves the accuracy of clustering results based on the protection of data privacy protection.

It can be seen from two algorithms' convergence time in Fig. 9 that the proposed improved algorithm converges is faster than traditional K-means and has fewer iterations.

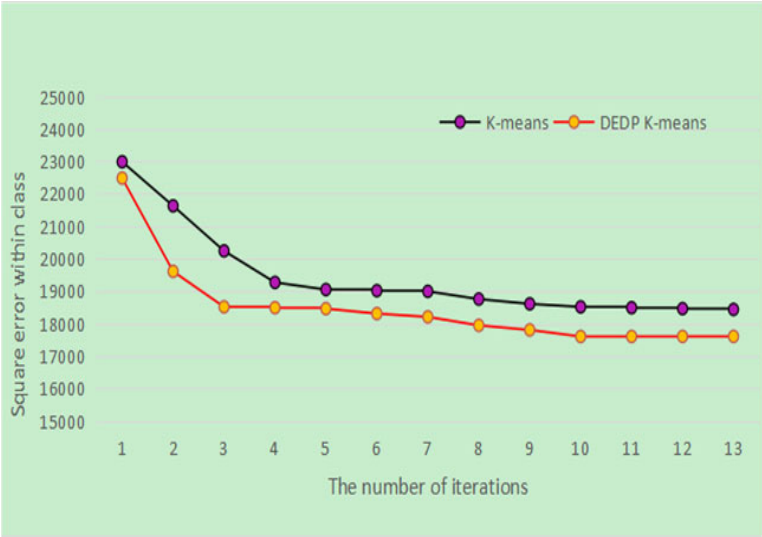

Fig. 9. Two kinds of algorithms' convergence time line chart

Fig. 10 records the results on Wine data set.

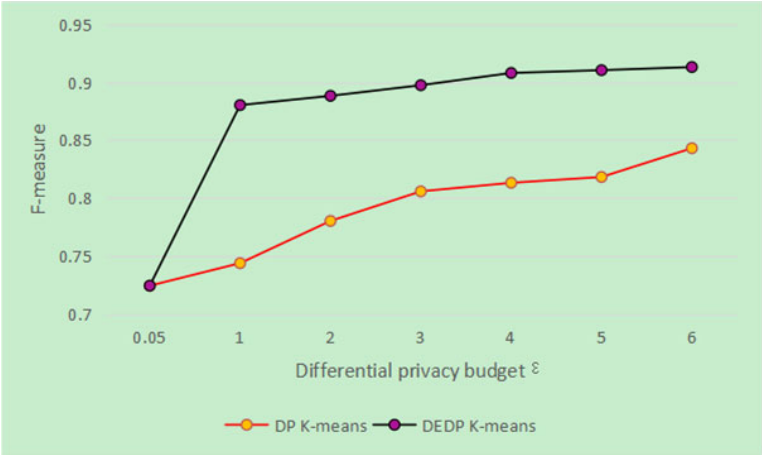

Fig. 10. The change of F-measure for Wine data set

Fig. 11 records the results on MAGIC data set.

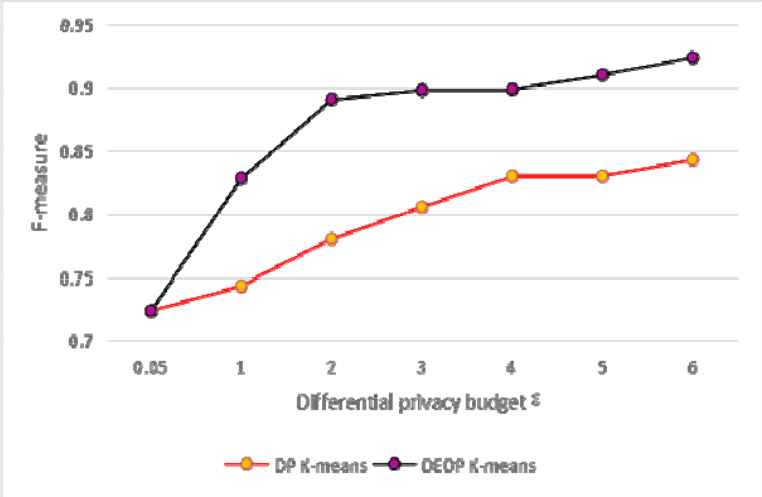

Fig. 11. The change of F-measure for MAGIC Gamma Telescope data set

It can be seen from Fig. 10 and Fig. 11 that for the same data set and ε, the F-measure of the DEDP K-means algorithm is higher than that of the traditional ε -differential privacy clustering algorithm, that is, under the same privacy protection budget, the clustering results proposed in this paper are more efficient. Compared with Fig. 10 and Fig. 11, it can be seen that the algorithm proposed in this paper is also applicable to large data sets, and the clustering results are more useful.

8 Simulation And Analysis—Cluster Parallelization Speedup Analysis

Speedup is defined in the case of fixed data size, by calculating the ratio of serial time and parallel time to determine the parallel effect, which is defined as $S = Ts/Tp$, the larger the S is, the better the parallelization effect is.

In this experiment, we choose the real physical clusters to build Spark computing platform, operating system is Ubuntu 14.04, Java version is JDK1.7.0-45, Hadoop version is 2.4.0, Spark version is 1.0.0. During the experiment, the running time of the Wine data set and the Magic data set on the single machine environment and the Spark platform is tested, the speedup is calculated and the results are analyzed.

As can be seen from the experimental results shown in Fig. 12, parallel algorithm can speed up the running speed. And with the increase of the number of nodes, the algorithm execution speed will be correspondingly accelerated. However, the speedup effect of different data sets on the Spark platform is not exactly the same, mainly in the difference of data size. In view of the massive data sets, with the increase of the number of nodes, the speedup generally increases linearly, but when increase to a certain amount, the speedup will no longer increase. Therefore, the number of nodes should be selected according to different data sets.

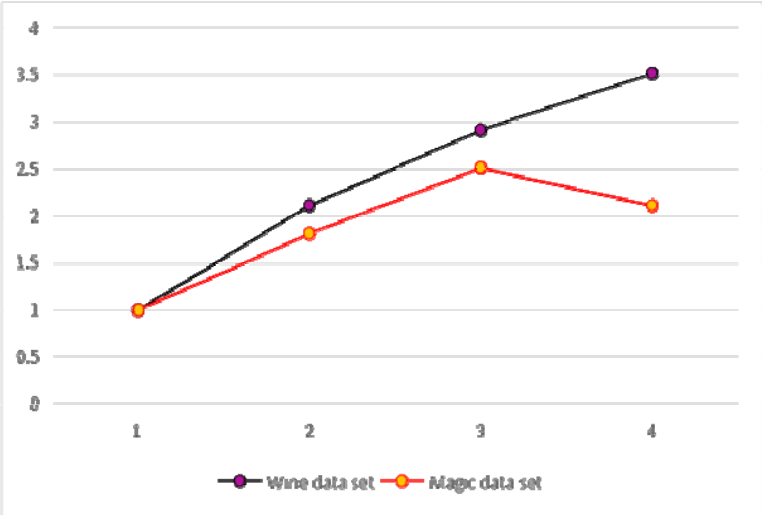

Fig. 12. Speedup of DEDP K-Means algorithm on Spark platform

9 Conclusion

The differential privacy protection model is widely used because of its strict mathematical theory. In big data mining, differential privacy technology solves the problem of data analysis and privacy disclosure. In this paper, a new privacy clustering algorithm -- DEDP K-means is proposed, in view of the traditional ε - differential privacy clustering algorithm is sensitive to the initial centers, and the differential evolution algorithm is introduced. As a swarm intelligence optimization algorithm, differential evolution algorithm is used to optimize the initial clustering centers by multiple iterations. In order to improve the efficiency of the algorithm, this paper introduces adaptive opposition-based learning technology, which uses the dynamic search boundary instead of the traditional fixed boundary, which is beneficial to search the neighborhood space and improve the local mining ability of the algorithm. At the same time, aiming at the problem of low efficiency in large-scale data sets, the improved algorithm is implemented on the

Spark platform in this paper. Through the test of two sets of data in the UCI data set, we can see that the clustering results of DEDP K-means algorithm proposed in this paper are more efficient and faster.

Acknowledgments

This research was supported by National Natural Science Foundation (31571563), Beijing Engineering Research Center of High Reliable Embedded System, Beijing Key Laboratory of Electronic System Reliability and Prognostics, the Project of Construction of Innovative Teams and Teacher Career Development for Universities and Colleges Under Beijing Municipality.

References

[1] C.C. Aggarwal, P.S. Yu, (Eds.), Privacy-Preserving Data Mining: Models and Algorithms, Springer, New York, NY, 2015.

[2] S. Zhou, F. Li, Y. Tao, Privacy preservation in database applications: a survey, Chinese Journal Of Computers 32(5)(2009) 847-861.

[3] L. Sweeney, K-anonymity: a model for protecting privacy, IEEE Security & Privacy Magazine 10(5)(2012) 1-14.

[4] L. Sweeney, Achieving K-anonymity privacy protection using generalization and suppression, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 10(5)(2012) 571-588.

[5] A. Machanavajjhala, J. Gehrke, D. Kifer, L-diversity: privacy beyond k-anonymity, ACM Trans. Knowl Discov Data 1(1)(2007) 24-35.

[6] C. Dwork, Differential privacy, in: M. Bugliesi, B. Preneel, V. Sassone, I. Wegener. (Eds.), Automata, Languages and Programming, Springer, Berlin, Germany, 2006, pp. 1-12.

[7] A. Blum, C. Dwork, F. Mcsherry, Practical privacy: the SuLQ framework, in: Proc. Twenty-Fourth ACM Sigact-Sigmod-Sigart Symposium on Principles of Database Systems, 2005.

[8] C. Dwork, A firm foundation for private data analysis, Communications of the Acm 54(1)(2011) 86-95.

[9] H. He, L. Guo, Y. Geng, The optimization of cmac neural network structure based on canopy-k-means algorithm, International Journal of Advancements in Computing Technology 4(22)(2012) 641-647.

[10] B. Fu, A. Zhang, K-means clustering text mining method using center estimation based on mean density, Journal of Chongqing University of Posts and Telecommunications(Natural Science Edition 26(1)(2014) 111-116.

[11] D. Fu, C. Zhou, Improved K-means algorithm and its implementation based on density, Journal of Computer Applications 31(2)(2011) 432-434.

[12] J. Xie, W. Guo, W. Xie, K-means clustering algorithm based on optimal initial centers related to pattern distribution of sample in space, Application Research of Computer 29(3)(2012) 888-892.

[13] D. Mao, Improved Canopy-Kmeans algorithm based on MapReduce, Computer Engineering and Application 48(27)(2012) 22-26.

[14] Y. Xu, W. Qu, Z. Li, Efficient K-means++ approximation with mapreduce, IEEE Transactions on Parallel & Distributed Systems 25(12)(2014) 3135-3144.

[15] N.L. Kazanskiy, P.G. Serafimovich, E.A. Zimichev, Spectral-spatial classification of hyperspectral images with k-means++ partitional clustering, in: Proc. Optical Technologies for Telecommunications. International Society for Optics and Photonics, 2015.

[16] G. Zhang, Research on a parallel genetic algorithm in hadoop and application in the site selecton of emergency facilities,

China Internet (8)(2013) 11-14.

[17] J. Dean, S. Ghemawat, MapReduce: simplified data processing on large clusters, in: Proc. Conference on Symposium on Opearting Systems Design & Implementation, 2004.

[18] A. Mcgregor, I. Mironov, T. Pitassi, The limits of two-party differential privacy, in: Proc. IEEE, Symposium on Foundations of Computer Science, IEEE Computer Society, 2010.

[19] J. Macqueen, Some methods for classification and analysis of multivariate observations, in: Proc. the Fifth Berkeley Symposium on Mathematical Statistics and Probability, 1966.

[20] K. Price, R. Storn, K. Price, Differential evolution- a simple evolution strategy for fast optimization, Dr Dobb's Journal 22(4)(1997) 18-24.

[21] H.R. Tizhoosh, Opposition-based learning: a new scheme for machine intelligence, computational intelligence for modelling, control and automation, in: Proc. 2005 and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, 2005.

[22] S. Rahnamayan, H.R. Tizhoosh, M.M.A. Salama, Opposition-based differential evolution, IEEE Transactions on Evolutionary Computation 12(1)(2008) 64-79.

[23] H. Wang, F. Qian, Swarm intelligence optimization algorithm, Control and Instruments in Chemical Industry 34(5)(2007) 7-13.

[24] Y. Wang, L. Li, Z. Hu, Swarm intelligence optimization algorithm, Computer Technology And Development 18(8)(2008) 114-117.

[25] B. Wang, Improved artificial bee conlony algorithm based on local best solution, Application Research of Computers 31(4)(2014) 1023-1026.

[26] Z. Deng, Dunqian Cao, Xiaoji Liu, A new differential evolution algorithm, Computer Engineering and Applications 44(24)(2008) 40-42.

[27] J. Dean, S. Ghemawat, MapReduce: simplified data processing on large clusters, in: Proc. Conference on Symposium on Opearting Systems Design & Implementation, 2004.

[28] T. White, Hadoop: The Definitive Guide, Yahoo Press, New York, NY, 2010.