Classification Rules Mining Model with Genetic Algorithm in Cloud Computing

Jing Ding
School of Management
Hefei University of Technology
Hefei, China (230009)

Shanlin Yang
School of Management
Hefei University of Technology
Hefei, China (230009)

ABSTRACT
Cloud computing is a good platform for research and application of data mining, for the reason that it provides powerful capacities of storage and computing, excellent resource management based on virtualization and resource sharing model, and comprehensive service system. However, investigation on data mining in cloud computing environment is still in its infancy. In this paper, solvent of classification rules mining with resources in cloud is developed, and an innovative classification rules mining model with genetic algorithm in cloud computing is proposed considering characteristics of cloud computing. An illustrative example is analyzed to show feasibility and effectiveness of the suggested model.

General Terms
Cloud computing, Data mining, Classification, Distributed systems

Keywords
Data mining, Cloud computing, Classification, Genetic algorithm

1. INTRODUCTION
When cloud applications are popular and massive data has accumulated, data mining is an important issue for cloud services, such as Salesforce.com and YouTube. In order to improve quality of these cloud services, data mining on massive data is crucial, since customerised services are based on valuable information acquired by data mining. YouTube conducts recommendation by analyzing historical data and mining customers' interests. For Salesforce.com, data mining is a necessary technique to provide CRM service. Cloud computing has become a most popular research buzzword. Leading IT corporations such as Google, Amazon and IBM has proposed some cloud computing architectures. Some research institutes have also developed cloud computing platforms. For instance, in the Science Clouds Project initiated by University of Chicago and University of Florida, Nimbus Cloud and Florida Cloud were explored to provide lease resources to scientific community [1]. The intrinsic features of cloud can be concluded as: tremendous storing and computing capacities, scalable and flexible resources and structure, and on-demand service via virtualization and resource pool. These characteristics make it possible to implement data mining as a commercial application, and make data mining in cloud computing a research area valuable in theory and practice. Although cloud is excellent in storage and computation, it is also vital to have tools and environments that support analysis and discovery over these data.

Cloud computing is a good platform for data mining. It can offset defects of previous methods in analyzing network data. In cloud, resources for storage and computing were distributed. Thus, data mining in cloud is conducted radically different from the traditional mining operated on local computers, and meets the requirements of data mining in Internet. The infrastructure of cloud computing is constructed of considerable server clusters, which endow the cloud with powerful capacities of computing, storing, data analyzing and data management. These capacities provide essential preconditions to massive data mining on the Internet. In addition, IT resources and applications are provided as public facilities in cloud. As the way you use water, electricity and gas, you can use resources and application in cloud without considering where they come from and how to produce them. This is a service-oriented IT application model, which can be more adaptable to requirements of data mining development and application. Furthermore, according to the SaaS (Software as a Service) business model of cloud computing, data mining programs, software or platforms are packaged as a service and sold to users and developers. Enterprises can improve the scalability of their services and deal with bursts in resource

demands by employing cloud services [2]. This will help small and medium-sized enterprises reduce the cost of software development when implement data mining, and propagate business application of data mining consequently.

In the past few decades, parallel, distributed and grid techniques were applied to data mining. For parallel and distributed paradigms, database was divided into several segments, which were distributed to different computing nodes for data mining. By such a strategy, the global computational effort is shared. And the computing efficiency increases because the subtasks operate on distributed data sites concurrently [3]. Knowledge grid offers tools and techniques for distributed mining and extraction of knowledge from data repositories available on the grid [4].

Since data mining tasks become increasingly complex as data accumulating, research in the past few decades was focused on parallel and distributed mining techniques. In most of the research, database was divided into several segments, which were distributed to different computing nodes for data mining. By such a strategy, the global computational effort is shared. And the computing efficiency increases because the subtasks operate on distributed data sites concurrently [3]. However, the computing nodes will exchange transaction information among each other during the mining process. The high efficiency will be undermined by frequent and massive data interchanging. Meanwhile, information processing in network requires real-time communication. But parallel and distributed data mining do not guarantee excellent mechanism of information sharing and cooperation to fulfill such a significant requirement. In addition, the data privacy and security is also a major concern, since data may be illegally attacked when the parallel and distributed algorithms duplicates the database to every node [5]. In order to overcome these problems, researchers have launched investigation on data management and data analysis in cloud computing environment. Sakr et al. gives a comprehensive survey of numerous approaches and mechanisms of deploying data-intensive applications in the cloud, and discusses some open issues and future challenges pertaining to scalability, consistency, economical processing of large scale data on the cloud [6]. Cloud computing has opened up the challenge for designing data management systems that provide consistency guarantees at a larger granularity. Therefore, Agrawal et al. highlight some design principles for systems providing scalable and consistent data management as a service in the cloud [7]. In order to support the ECG data analysis, Pandey et al. design an autonomic cloud environment that collects health data and disseminates them to a cloud-based information repository and facilitates data analysis using software services in the cloud [8]. Although data management and analysis in cloud have been explored in depth, research focused on data mining in cloud is not enough. Issues, such as algorithm and system architecture of data mining in cloud, need further investigation.

However, data mining in cloud computing environment is not a novel field. It can be implemented in cloud according to some traditional methodologies. Some drawbacks of data mining can be undermined when exploited in cloud. The challenge here is how to adapt existing data mining models and techniques into the cloud. Hence, in this paper, we exploit the cloud computing environment naming Cloud based Genetic Classification Rules Mining Model (CGCRMM) to address classification rules mining problem. The procedure of classification is arranged considering the distributed and parallel cloud environment. And the adapted genetic algorithm, which makes good use of the computing power of cloud computing, is designed to solve this model. For training and testing the proposed model, we use data collected from UCI dataset to conduct an illustrative example. Rest of the paper is structured as follows: Section 2 is a brief review of the literature relevant to data mining in cloud computing environment; Section 3 describes the basic methodologies and detailed construction of the CGCRMM model; Experiment in section 4 evaluate the validity and performance of the proposed model; Section 5 concludes the whole research.

2. RELATED WORKS

Distinguishing with the traditional mining paradigms, data mining in cloud is a novel area filled with valuable issues worthy of investigation. Basing on an intensive review on the relevant literature, most of the researchers concentrate on the following problems.

2.1. Data Mining Algorithm

Cloud computing, with its promise of virtually infinite computing and storage resources, is suitable to solve resource greedy computing problems. One problem of data mining in the cloud has been investigated from the data mining algorithm perspective. Wang et al. [9] utilized the powerful and huge capacity of cloud computing into data mining and machine learning. In their experiments, three algorithms, i.e., global

effect (GE), K-nearest neighbor (KNN) and restricted boltzmann machine (RBM) were performed in cloud computing platforms, which use the S3 and EC2 of Amazon Web Services. And they built two predictors based on KNN model and RBM model respectively with the order to testify their performance based on cloud computing platforms.

The MapReduce programming model was designed for processing massive data sets in a parallel network. Based on this programming model, Wang et al. [10] adapted the SPRINT algorithm which is ideal tool for data classification. SPRINT has been designed to be easily parallelized. Due to the parallelism, the original SPRINT was modified to be implemented in Hadoop architecture. The algorithm divided datasets in vertical direction and horizontal direction respectively, in accordance with the "Map" step in MapReduce. The vertical partition separated datasets by attribute, while horizontal partition produced many item sets. They applied the revised SPRINT algorithm to classify customer groups with different credit grades.

2.2. Architecture of Data Mining

Cloud is an infrastructure that provides resources and services over the Internet. Generally speaking, a cloud computing platform consists of a storage cloud, a data cloud and a compute cloud, which are responsible for storage services, data management and computational tasks. Google's App Engine platform is composed of Google File System (GFS), BigTable and MapReduce [11]. Amazon provides its cloud services by Amazon Web Service (AWS), which contains Simple Storage Service (S3), Simple DB and Elastic Computing Service (EC2) [12].

Grossman et al. [13] developed a cloud-based infrastructure to support data mining applications. The infrastructure consists of a storage cloud called Sector and a compute cloud called Sphere. The Sector storage cloud was designed for wide area, high performance networks and employed specialized protocols to utilize the available bandwidth, while GFS and Hadoop Distributed File System (HDFS) assume small bandwidth and don't work well with loosely coupled distributed environments. The Sphere compute cloud allows user defined functions.

3. METHODOLOGIES AND CGCRMM MODEL

3.1. Classification

Classification is an important mission in data mining, and probably has become the most studied data mining task. In this task, the goal is to predict the value of a specified goal attribute (called the class attribute) based on the values of other attributes (called the predicting attributes) [14]. The dataset is symbolized by an attribute set consists of a number of attributes, R=(a_1, a_2,…, a_N), where a_i(i=1,2,…,N) is an attribute. The attribute set can be divided into two parts: 1) predicting attributes, C=(c_1, c_2,…, c_m); 2) class attributes, D=(d_1, d_2,…, d_n). A classification rule is demonstrated as:

If $(c_1 \in I_1) \wedge (c_2 \in I_2) \wedge \ldots \wedge (c_m \in I_m)$,

Then $(d_1 \in J_1) \wedge (d_2 \in J_2) \wedge \ldots \wedge (d_n \in J_n)$

where I_i and J_j (i=1,…,m; j=1,…,n) are values of c_i and d_j respectively. "If" part contains a combination of conditions, and "Then" part contains the predicted class labels.

The data record is horizontally divided into two parts – training set and test set – that are mutually exclusive and exhaustive. The data mining algorithm discovers classification rules based on the training set, and evaluates the predictive performance of these rules with the test set.

The application of algorithm to a data set can be considered the core step of classification. The commonly used classification methods include: Bayesian reasoning, decision tree, genetic algorithm, rough set, case-based reasoning, neural network et al.

3.2. Genetic Algorithm

Genetic algorithm (GA) is a computational model inspired by evolution. This algorithm encodes a specific problem with several attributes on a simple chromosome-like data structure, and generates a population of chromosomes randomly as the initial colony. Then it applies operators - selection operator, crossover operator and mutation operator - to these structures to get an optimal solution evaluated by a fitness function. The workflow of GA is depicted as follow:

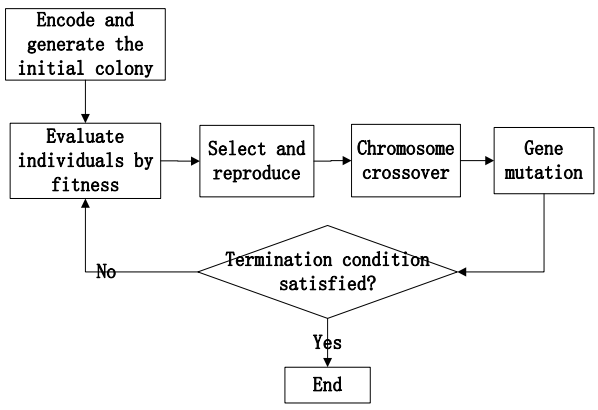

Fig.1 Workflow Chart of Genetic Algorithm

In contrast to most data mining algorithms, GA is a simple, robust, easy-to-operate and global-search method. By crossover and mutation operations on individuals, a variety of individuals can be generated. This diverse population will reduce probability of performing a local search. In addition, genetic operations proceeding randomly according to the probability will require no manual effort to guide and adjust the algorithm. Furthermore, GA is a more efficient approaches, because the fitness function is designed corresponding to the goal of mining task. These factors motivate us to use GA as a resolution for classification rules mining.

3.3. Classification Rules Mining in Cloud Computing Environment

3.3.1 Constraints analysis

Cloud computing is a network where distributed computer clusters constitute the resource pool of hardware. Tasks are divided into parallel segments, and assigned to available computing resources for processing. In that case, whether a computing task can be handled by cloud depends on decomposability and parallelism of the task. The requirements of processing computation tasks in cloud are defined as follow:

a) The task is capable of being divided into mutually exclusive sub-tasks.

b) Sub-tasks and data should be allocated to unoccupied processing nodes.

c) Mechanism of synchronization and communication amongst processing nodes is indispensable.

GA, proposed by Prof. J. Holland in University of Michigan in 1975, was derived from evolutionism based on Darwin's theory of genetic selection and nature elimination. It is characterized by parallelism and global searching. During the procedure of GA, no dependency emerges among individuals, due to the genetic operations on them are determined only by coding and fitness of individuals, which are independent factors. Fitness evaluation of an individual is independent of the evaluation of any other individual. Evolution of different sub-populations carries out synchronously. It is noticeable that GA has good performance on parallelism inherently, and will be well adapted for use in cloud. Consequently, the classification rules mining model is proposed based on GA.

3.3.2 Model description

With storage and computing resources dispersing in the cloud environment, a Master is set as a central controller to search and distribute the resources. The responsibilities of Master are defined as: First, it divides the entitle task into several sub-tasks (here, a sub-task can be regarded as a data section), and assigns them to dispersed processing node; Afterwards, sub-tasks are executed on distributed nodes under the supervision of the Master; Eventually, the Master collects processing results on every node and synthesizes them to present a comprehensive set of classification rules as the mining result.

The entire procedure of classification rules mining consists of two stages: rules detection and classes induction. In the first stage, a Processing Unit (PU) deals with a sub-task via translating each piece of data into a statement in form of "If … Then …" and detecting rules within the scale of sub-task. The results are raw rules. And in the second stage, Class Builders (CB) implement induction on the results from PUs. A CB induces rules with a certain class by analyzing the raw rules sharing this class attribute. Results of CBs are generalized as a rule set by Master and presented to users. Task distribution and coordination among PUs and among CBs are administered by the Master.

The classification rules mining model is displayed by Fig. 2 and the operating principle of it is explained explicitly as:

a) A mining task is requested to the Master. The Master searches available nodes which are not occupied by computation task, and set them as PUs. Then the Master divides the data file into data segments according to the quantity of PUs and the file size. If there are n nodes available and the data file size is measured as m, the size of every data segment is defined as m/n.

b) Master sends a copy of a data segment to every PU.

c) PU translates every piece of data recording into a genetic chromosome according to the GA encoding. Based on GA, every PU implements rules detections on the data segment it possesses. The detection results are stored in a buffer and expressed in format of <key, value>, where key stands for class attributes and value stands for predicting attribute.

d) Master reads the data in buffer periodically and generates a list of <key, valueList> recordings, where key is the class

attribute and valueList is a set of rules sharing the same class attribute.

e) When all the PUs finished their work, Master releases computing resource occupied by PUs.

f) Master searches available nodes to establish CBs, the quantity of which is the number of key value in the list of <key, valueList>. Then the Master distributes every CB with a piece of record in <key, valueList>, which is a set of rules sharing a certain class value.

g) On the basis of GA, CB generates classification rules for a certain class value by implementing rules reduction on the rules set belongs to it, and transmits the results to the Master.

h) Master collects processing results from CBs and synthesizes them to present a comprehensive set of classification rules as the mining result.

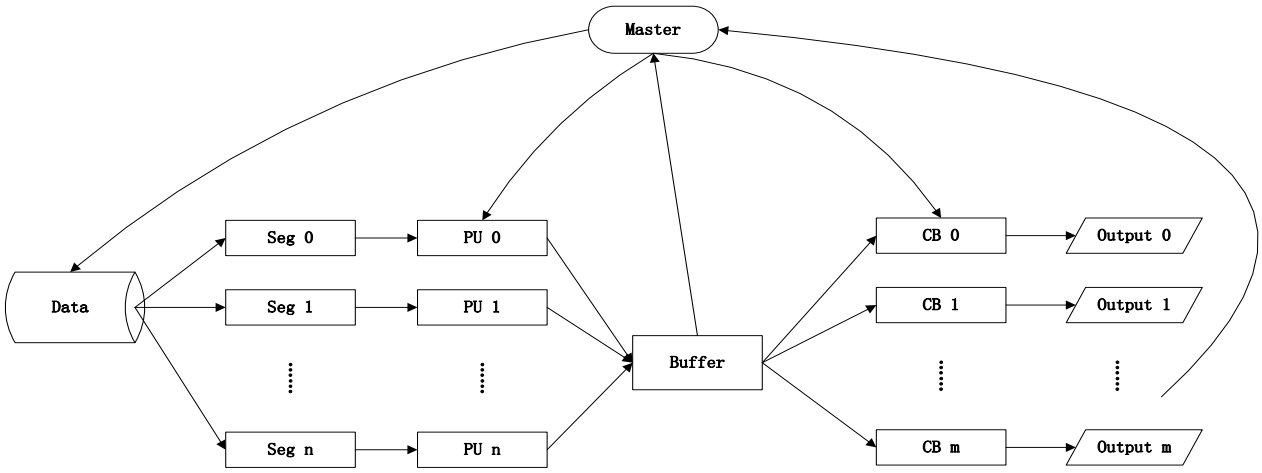

Fig.2 Classification Rules Mining Model in Cloud Computing Environment

The pseudo code of programming the model is displayed as:

```
SegSet=Master.dataSeg(DateFile, N);
SegSet=(Seg(1), Seg(2),…, Seg(N));
For(i=1;i≤N;i++)
{PUData(i)=Master.copyData(Seg(i));
GAClassify(PUData(i));
//calculate classification rules in form of <key, value>
Input classy results into Buffer;}
For(j=1;j≤m;j++)
   key(j).valuelist=Master.calculateValuelist(key(j));
   //for each key in buffer, calculate a valueList consists of
   rules sharing this key.
For(j=1;j≤m;j++)
{CUData(j)=Master.distributeValuelist(ValueList(j));
GAClassify(CUData(j));
//calculate classification rules in form of <key,
valueList>}
Master.getResults(CU);
```

3.4. Algorithm Design

The canonical GA is adapted to cloud computing environment for mining classification rules. However, it follows the basic procedure of canonical GA, including encoding, initialization, fitness assessment, selection, crossover and mutation.

3.4.1 Encoding

The first step of GA is encoding that represents variables of individuals into bit strings. Pittsburgh and Michigan approaches can be applied to encode individuals. In the Pittsburgh approach each individual encodes a set of prediction rules, whereas in the Michigan approach each individual encodes a single prediction rule. When the task is classification, interaction among the rules is important. The Pittsburgh approach seems more natural to evaluate the quality of the rule set as a whole. However, by evaluating each rule independently of other rules, the Michigan approach might be more natural in the task where the goal is to find a small set of high-quality prediction rules [14]. Therefore, the Michigan approach is adopted for encoding in this paper. An individual represents a single rule by expressing its attributes with binary strings. Suppose that a given attribute can take on k discrete values. Then we can encode value of this attributed by using k bits. However, if the attribute is continuous, the continuous value range should be discretized into several discrete values.

For instance, suppose that the attribute values of Marital_Status can be “single”, “married”, “divorced” and “widow”. Then the attribute value would be encoded in the genome by four bits. For example the value “0 1 1 0” would represent the condition “Marital_Status= “married” or “divorced”.

3.4.2 Initial population and fitness function

Pieces of data in the dataset are encoded into binary strings and denoted as the initial population. After creating the initial population, evaluate each string by a fitness function with the result of a fitness value.

The fitness function transforms performance of an individual into an allocation of reproductive opportunities, according to predicting and class attributes of a rule. The fitness function is constructed based on support and confidence. Suppose the population is N, and the rule is “If A Then C”. Then some symbols are defined:

TT is the quantity of “If A Then C”;
TF is the quantity of “If A Then Not C”
FT is the quantity of “If Not A Then C”
FF is the quantity of “If Not A Then Not C”
Then the fitness of the rule “If A Then C” is calculated as:
Support= TT/N; Confidence = TT/(TT+TF);
Fitness=confidence+support=TT/N+TT/(TT+TF)

The fitness function is a combination of support and confidence, which are two vital indexes evaluating a string. The support value is the proportion of a positive rule. A rule with big support value is more prevalent in the population, and has more probability to be true. The confidence value defines possibility of a certain class attribute when classify strings with certain predicting attributes. It is a significant factor indicating the validity of a rule.

3.4.3 Selection operator

The “remainder stochastic sampling” [15], which will more closely match the expected fitness values, is applied to conduct selection process. For each individual *i* where $f_i / \overline{f}$, the integer portion of this number indicates how many copies of individual directly placed in the next generation, and the decimal portion represents the chance of placing another copy in offspring.

3.4.4 Crossover operator

After selection operation, crossover occurs. Crossover is recombination of paired strings in order to create new samples. According to the probability of P_c, a crossover point is randomly chosen. Then the fragments after the crossover point swap and produce the offsprings.

3.4.5 Mutation operator

When crossover finished, it is time for mutation. Mutation is applied to each bit in the strings with a low probability denoted P_m. If a bit is selected to mutate, the bit value will change within its value range.

After the process of selection, crossover and mutation, the next generation is formed. The pseudo code of GA designed for classification rules mining is displayed as follow:

Begin t=0;
Initialize P(t)={$X_1(t), X_2(t), \ldots, X_n(t)$};
Fitness(P(t))={fitness($X_1(t)$), fitness($X_2(t)$), …, fitness($X_n(t)$)};
While(fitness(P(t))<1- ε and t<maxGeneration){
Selection(t)=Select(P(t));
Crossover(t)=Cross(P(t));
Mutation(t)=Mutate(P(t));
P(t+1)={Selection(t) ∪Crossover(t) ∪Mutation(t)};
t++;}
End

4. EXPERIMENT

In order to evaluate the validity of the proposed model, an illustrative example is applied to examine the performance. Data of breast cancer symptom, collected from the UCI dataset [16], are used as the raw data of our experiment. The dataset includes 699 pieces of data which are characterized by 11 mutually exclusive attributes. A portion of the dataset is shown in Tab.1. Two thirds of the data are defined as the training set, and the rest is left for test set.

Tab.1 Wisconsin Breast Cancer Database

Sample code	Clump thickness	Uniformity of cell size	Uniformity of cell shape	Marginal adhesion	Single epithelial cell size	Bare nuclei	Bland chromatin	Normal nucleoli	mitoses	class
63375	9	1	2	6	4	10	7	7	2	malignant
76389	10	4	7	2	2	8	6	1	1	malignant
95719	6	10	10	10	8	10	7	10	7	malignant
128059	1	1	1	1	2	5	5	1	1	benign
142932	7	6	10	5	3	10	9	10	2	malignant
144888	8	10	10	8	5	10	7	8	1	malignant
145447	8	4	4	1	2	9	3	3	1	malignant
160296	5	8	8	10	5	10	8	10	3	malignant
167528	4	1	1	1	2	1	3	6	1	benign
183913	1	2	2	1	2	1	1	1	1	benign
183936	3	1	1	1	2	1	2	1	1	benign
188336	5	3	2	8	5	10	8	1	2	malignant
191250	10	4	4	10	2	10	5	3	3	malignant
242970	5	7	7	1	5	8	3	4	1	benign
255644	10	5	8	10	3	10	5	1	3	malignant
263538	5	10	10	6	10	10	10	6	5	malignant
274137	8	8	9	4	5	10	7	8	1	malignant
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

Periods of the cancer corresponding to different patients are classified with the proposed model based on their symptom. By comparing the result derived from the classification model with diagnosis concluded by the doctors, the validity of the proposed model could be assessed. Classification accuracy and time consumption are two indexes indicating efficiency of the model.

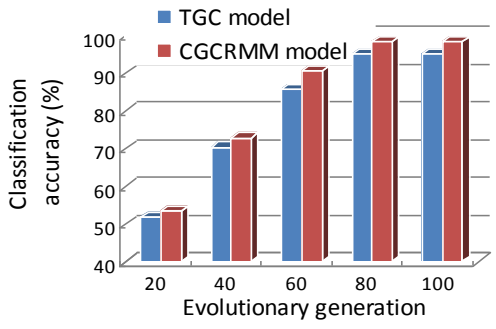

Fig. 3 Classification accuracy of CGCRMM model

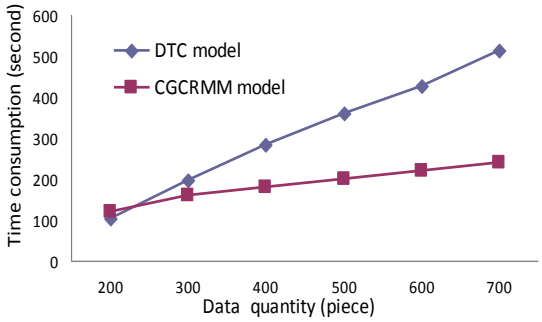

Fig. 4 Time consumption of CGCRMM model

From the results displayed in Fig. 3, it is notable that with the same evolutionary generation, the proposed model is more accurate than the traditional genetic classification (TGC). And when the population evolves more than 60 times, the classification accuracy will be much higher than the TGC model.

In Fig. 4, execution time with different data quantity is shown. CGCRMM is compared with decision tree classification (DTC). When handling a small amount of data, for example less than 250 pieces of data, the proposed model is not superior to the DTC model and even has a lower efficiency, because data

transmission offset the time saved by distributed work. This is also a phenomenon in cloud computing that task with light load could hardly take advantage of the powerful computation capability in cloud. However, time consumption of the proposed model increases much slower than the increase of data amount. And the proposed model requires far less time than DTC model when the data is massive. It can be concluded that the CGCRMM model is high efficient when tackling with a huge number of tasks.

5. CONCLUSION

Cloud computing is characterized by its powerful capability of computation and storage, as well as its policy of resource sharing accomplished by virtualization. These features render cloud computing valuable merits favorable to data mining service in network environment. According to this, we developed solvent of classification rules mining with resources in cloud. A classification rules mining model with genetic algorithm in cloud computing is proposed considering characteristics and key techniques of cloud computing. An illustrative example is analyzed to show feasibility and effectiveness of the suggested model. The integration of classification and cloud computing in this paper is at the initial stage of our research on data mining service in cloud and requires further improvement.

6. ACKNOWLEDGMENTS

Our research is funded by the Chinese Government through the National Natural Foundation programs (No.71071045 and No.71131002).

7. REFERENCES

[1] Keahey, K., Figueiredo, R., Fortes J., Freeman, T., and Tsugawa, M. 2008. “Science Clouds: Early Experiences in Cloud Computing for Scientific Applications”. In Proceeding of High Performance Computing and Communications.

[2] Buyya, R., Yeo, C. S., Venugopal, S., Broberg, J., and Brandic, I. “Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility”, Future Generation Computer Systems, 25(6), pp.599-616, 2009.

[3] Park, B. H., and Kargupta, H. 2002. “Distributed data mining: algorithms, systems, and applications”, Data Mining Handbook..

[4] Cannataro, M., Talia, D., and Trunfio, P. 2002. “Distributed data mining on the grid”, Future Generation Computer Systems, 18, 1101-1112.

[5] Lin, K. W., and Luo Y. C. 2009. “A fast parallel algorithm for discovering frequent patterns”. In Proceeding of IEEE International Conference on Granular Computing.

[6] Sakr, S., Liu, A., Batista, D. M., and Alomari, M. 2011.“A survey of large scale data management appproaches in cloud environments”, IEEE Communications Surveys & Tutorials, 13(3), 311-336.

[7] Agrawal, D., Abbadi, A. E., Antony, S., and Das, S. 2010. “Data management challenges in cloud computing infrastructures”, Databases in Network Information Systems, 5999, 1-10.

[8] Pandey, S., Voorsluys, W., Niu, S., Khandoker, A., and Buyya, R. 2012. “An autonomic cloud environment for hosting ECG data analysis services”, Future Generation Computer Systems, 28(1), 147-154.

[9] Wang, J., Wan, J., Liu, Z., and Wang, P. 2010. “Data mining of mass storage based on cloud computing”. In Proceeding of 2010 Ninth International Conference on Grid and Cloud Computing.

[10] Wang, E., and Li, M. 2009. “Research on mass data mining under cloud computing”, Modern Computer, 22-25.

[11] Bedra, A. 2010. “Getting started with Google App Engine and Clojure”, Internet Computing, 14, 85-88.

[12] Wang, L., Tao, J., Kunze, M., Castellanos, A. C., Kramer, D., and Karl, W. 2008. “Scientific cloud computing: early definition and experience”. In Proceeding of the 10th IEEE International Conference on High Performance Computing and Communications.

[13] Grossman, R. L., Gu, Y., Sabala, M., and Zhang, W. 2009. “Compute and storage clouds using wide area high performance network”, Future Generation Computer Systems, 25, 179-183.

[14] Freitas, A. A. 2002. “A survey of evolutionary algorithms for data mining and knowledge discovery”, Advances in Evolutionary Computation, 819-845.

[15] Booker, L. B., Goldberg, D. E. and Holland, J. H. 1989. "Classifier systems and genetic algorithms", Artificial Intelligence, 40, 235-282.

[16] William H., Wolberg, W., Street, N., and Mangesarian, O. L. 1992, UCI Machine Learning Repository, http://archive.ics.uci.edu/ml/datasets.html.