Research on Big Data Query Online Analysis and Processing Technology on the Basis of Decision Tree Model

Lixia Liu[1,2,*], Hong Mei[1] and Bing Xie[1]

[1]*School of Electronics Engineering and Computer Science, Peking University, China*

[2]*Department of Information Engineering, Engineering University of CAPF, China*

Abstract: In this paper, we prompt a new big data query online analysis and processing method based on decision tree model. The purpose of this dissertation is to realize the high-efficiency of multi-dimensional OLAP query by studying the key approaches for improving the OLAP query efficiency. In order to bring the full play of data analysis, the concept of on-line analytical mining (OLAM) is used for reference in this dissertation. Data mining techniques and statistical analysis approaches are integrated to form an OLAP query frame. Then the key techniques are studied in the framework. A new kind of OLAP query, which is different from traditional cube query, is proposed, called multi-boid query. This query can be used in the moving objects network. And we implement graph cubes through the combination of moving objects network characteristics and existing data cube technology.

Keywords: Decision Tree Model, Online Analysis and Processing Technology, Big Data, Query.

1. INTRODUCTION

As one of the key techniques of business intelligence (BI), on-line analytical processing (OLAP) is a very important way for knowledge acquisition and decision support. Meanwhile, due to the increase of data volume and data dimensionality caused by the improvement of information, and the high-efficiency required by decision support, the research on efficient OLAP query is developing. Aiming at decision support, it becomes an important research topic to study how to improve the multi-dimensional OLAP query efficiency in massive high-dimensional data sets so as to shorten the query time [1].

OLAP design method is used to realize the high-efficiency of multi-dimensional OLAP query by studying the key approaches for improving the OLAP query efficiency. In order to bring the full play of data analysis [2, 3], the concept of on-line analytical mining (OLAM) is used for reference in this dissertation. Data mining techniques and statistical base of OLAP query of which the construction way directly influences on the OLAP query efficiency, the materialization way of data cube is studied. On the other hand, OLAP approximate query approach is capable to realize the tradeoff between the query time and the query accuracy, which is beneficial to remarkably improve the OLAP query efficiency, so it is also considered to be a part of main content of this dissertation. Meanwhile, in order to improve the efficiency of OLAP query, the recommendation of OLAP query dimensions is another way from the other perspective. This approach focuses on aid decision making, which provides the users the dimensions closely related to the query target so as to shorten the query time.

In distributed systems, commonly used scheduling scheme includes centralized scheduling and distributed scheduling scheme. In the centralized model, the scheduler maintains surface information about all servers. AIT' jobs are submitted to the scheduler, the scheduler makes scheduling decisions. The centralized scheme is easier for implementation, but it is not very robust [4]. In the distributed model, each server maintains a scheduler and a job' queue, and they are self-managed. So although it is more robust, yet is complex for implementation [5].

The sender-initiated algorithm is easier and used generally in distributed systems. In the distributed OLAP system, there may be a number of enquiries tasks at the same time. When a task is complicated, system should decompose it into several subtasks. So, in the paper, we propose a hybrid distributed scheduling scheme with centralized scheme and stable sender-initiated algorithm based on common models. In this scheme, when the local server gets a query task, firstly the system uses centralized algorithm to allocate the task, and then the other sites use the stable sender-initiated algorithm to adjust Load, so the high load sites can assign their task to others [6].

With the rapid development of the Internet of Things technology such as RFID and wireless LAN, a large number of moving objects data is produced. These moving objects generally have multi-dimensional attributes, and the spatial and temporal characteristics. These moving objects communicating with other moving objects generate moving objects network. How to handle and effectively use the massive data generated by these moving objects, and apply ware-

*Address correspondence to this author at the School of Electronics Engineering and Computer Science, Peking University, China;
Tel: 13709117282; E-mail: wjllx939@tom.com

1874-4443/15 2015 Bentham Open

house and on-line query analysis to the moving objects network become one of the research focuses.

Considering that the moving objects network contains very specific valuable substructure information, there are many structured data query algorithms, but most of these algorithms are designed for static data. However, these algorithms are very difficult to play a role in processing the moving objects data generated by the moving objects. In this article, we study the on-line analysis and processing of the structured data on the moving objects network. Another important issue is how to build a moving object graph cube generated by a large number of aggregate graphs which is unlike traditional numeric data. These aggregate graphs contain a plurality of nodes and have a complicated structure, which requires a lot of storage space. And it will take more time to process and analysis these aggregate graphs data. How to compress so many aggregate graphs data has become an urgent problem. In this paper we conduct some in-depth research on the moving objects data, efficiently compress the nodes and edges of the aggregate graphs, and provide the effective help for the users to analyze the moving objects data [7].

2. THE SCHEDULING SCHEME OF ONLINE ANALYSIS AND PROCESSING SYSTEM

In a distributed system, how to design the scheduling scheme depends on the actual state of the system. The purpose of the scheme is how to meet the need of the system while reducing the complexity of the system. In the Distributed OLAP system, the design of scheduling scheme is mainly considered with the following aspects:

In a distributed system, how to design the scheduling scheme depends on the actual state of the system. The purpose of the scheme is how to meet the need of the system and reduce the complexity of the system. In the Distributed OLAP system, the design of scheduling scheme is mainly considered with the following aspects:

1) Certainty and enlightening algorithm

Certainty algorithm is used when all acts of the process can be forecasted. But we cannot forecast the system load, since the load conditions change all the time. So enlightening algorithm is adopted for the distributed OLAP system.

2) Centralized and distributed algorithm

Centralized algorithm is easy for implementation, yet it is not very robust. With the distributed algorithm, each site is self-managed. It can dynamically adjust the load of the system in the distributed OLAP system. When a query task appears, firstly system uses centralized algorithm on local server, while each server can adjust the load. When a server is overloaded, it can assign the new task to others.

3) Sender-initiated and receiver-initiated algorithm

When a site wants to assign a task, it must ascertain that the site that it sends the task to. It needs the load information of other sites to decide which server should be sent the task to. There are two paths to deal with the question: one is initialized by the sender, the other is initialized by the receiver. Sender is overload, it will send the task to others, whereas receiver is the low load site, and it can receive tasks from others. In the sender-initiated algorithm, to obtain load balance, the sender always tries to send the tall when it is overload [8]. The sender-initiated algorithm is easy to be understood and to be designed, so it is adopted for the distributed OLAP system.

In the distributed OLAP system, at a moment, there is only one coordinator. When the server receives a new query task, it assigns it as centralized algorithm. Firstly it will check its own load condition, if the load of the server is little then the scheduler will assign the task to itself, else it will assign the task to the lowest load server accordingly. Because the system is changed dynamically with performance of tasks, the load condition of servers is also changed, while the server updates its global load table per a certain period time, so the load of server that receives the task from other servers may not be the lowest. In the worst condition its load may be the highest. Thus, when a server whose load is higher receives a new task, it should assign the task to a low load server. In the system, the scheduler use stable sender-initiated algorithm to assign tasks.

On each server there are four queues: Sender queue, Ok queue, Receiver queue and Mission queue. The first three queues are used to identify the status of each server; the Mission queue is used to store tasks. The system defined unified threshold loads LT and UT; LT means lower limit and UT means upper limit. The server state depends on its load conditions. In the system, we can get the load condition from the four queue length.

Receiver queue: load of the server<LT

Ok queue: LT<=.load of the server <=UT

Sender queue: load of the server>UT

Servers in the Receiver queue have low load; they can continue to receive task. Servers in the Ok queue have moderate load; they can just deal with the received tasks. Servers in the Sender queue have high load; they cannot receive new tasks. If these servers receive new tasks, they should assign the task to other severs. When a server receives a new task, it should firstly check the site state after receiving the task. If the state is "Receiver" or " Ok ", the server should receive the task and insert it into its Mission queue. If the state is "Sender", the server should assign the task to other servers. For assigning the task, the server needs to ascertain that a server that can receive the new task. Firstly, the scheduler randomly finds a server from Receiver queue and check if it can receive the new task. If the server is not busy and it can receive task then assign the task to it, else it will find next server from the Receiver queue to check. Just do the same checking until finding a moderate server. If there are no moderate servers in the Receiver queue, scheduler then inserts the task into its own Mission queue. Fig. (**1**) shows the schedule scheme of the system.

There is a Mission queue on each server of distributed OLAP system. On the server all received query tasks are treated equally. A new task will be inserted into the end of Mission queue.

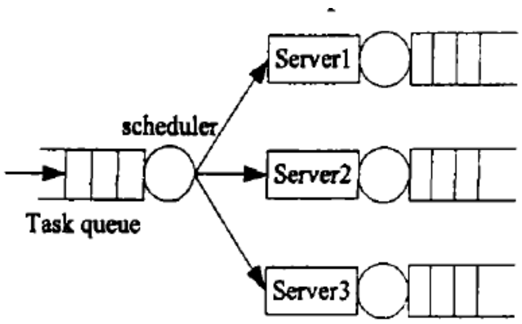

Fig. (1). The schedule scheme of the system.

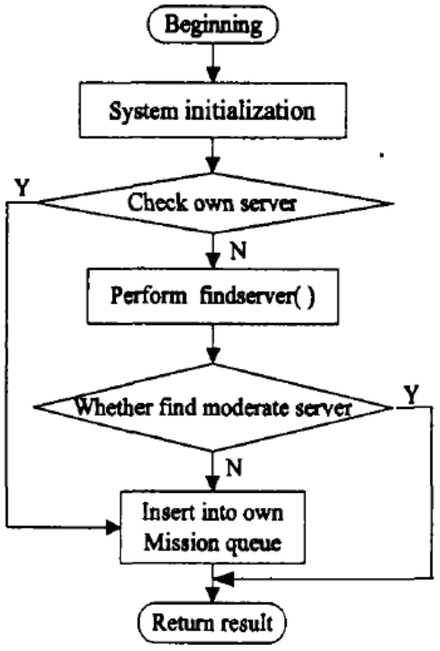

Fig. (2). The flowchart of a scheduling course.

According to the real system design, the distributed job scheduler is divided into three modules: initialization, server status updating and scheduling module. When the distributed system restarts, all servers begin to initialize the scheduler. First, it numbers the servers from node i to node n, and initializes some global variables, and then it waits for a new task. At a moment there is only one scheduler enabled. When the server receives a new task, it inserts the task into its Mission queue, and schedules the other tasks. The scheduler updates the servers' information through the status updating module. The communication among servers completes through the Windows Message Processing System. One scheduling process is shown as Fig. (**2**).

When the scheduler receives a new task, it checks itself firstly, if it can receive the task, then the task will be performed on the local server. If the server will be a member of the Sender queue after receiving the task, it should send message to the servers of the Receiver queue, and then assign the task according to the responses of these servers. If all the servers in Receiver queue cannot receive the new task, then the task will be inserted into the local Mission queue.

When a server receives a new task, if its load is low, the server will insert the task into Mission queue. Then the load of the server is changed, and the status may also be changed, so it updates its own status. If its status is changed from low to high load or from high to low, it should send messages to other servers.

3. THE ALGORITHM OF GENERATING DECISION TREE

In a learning problem based on the information entropy minimization heuristic, the attributes may be nominal-valued or continuous-valued in general. The continuous-valued attributes must be discretized prior to attributes selection or be fuzzified into linguistic terms. Here, we propose an algorithm for handling the fuzzy number-valued attributes in a classification problem.

Suppose $\left(A_a\right)>T$ and suppose $\left(A_a\right)$ not more that T. Suppose

$$\left(A_a\right)=\left[x,y\right]\left(x,y\in R\right) \tag{1}$$

The data to be used in equation (1) is normalized.

$$\overline{x_i}=\frac{x_i-b_i}{a_i-b_i} \tag{2}$$

Where X$_i$ and X$_j$ respectively, are the actual value and Standard value of the i-th index; a$_i$, b$_i$ are respectively the maximum and minimum of the i-th index. Depending on problems and experimental data, known evaluation indexes m and n are hidden layer nodes are determined, in order to find the value of the formula (3).

$$n=\log_2 m \tag{3}$$

Hidden node output is calculated as follows:

$$h_j=f\left(\sum_{i=1}^{m}w_{ij}x_i-\theta_j\right) \tag{4}$$

Where θ$_j$ is defined as the threshold value for hidden node.

The output of the output node is calculated as follows:

$$f\left(\sum_{i=1}^{m}w_{ij}x_i-\theta_j\right)=f\left(f\left(\theta_j\right)\right) \tag{5}$$

Where in θ is an output node threshold.

Equation (3) and Equation (4) in the transfer function is generally expressed as (0,1) interval of S-type function:

$$f\left(\sum_{i=1}^{m}w_{ij}x_i-\theta_j\right)=f\left(f\left(\theta_j\right)\right) \tag{6}$$

Using decision tree generating algorithm to train, the neural network can be used in the mechanical strength experiment on emulsified asphalt cement-stabilized macadam. The input layer connection weights can be denoted as w_{ij}, hidden layer and output layer connection weights can be denoted as t_j, hidden layer threshold can be denoted as θ_j and the output layer threshold can be denoted as θ. So connected together to form into a long string (the string corresponding to each position of a group of network weights and threshold value), an individual is constituted. It can generate an initial population of N individuals.

$$E\left(A,T,S\right)=\frac{\left|S_1\right|}{\left|S\right|}E\left(S_1\right)+\frac{\left|S_2\right|}{\left|S\right|}E\left(S_2\right) \tag{7}$$

where $N=\left|S\right|$ is the number of examples in the set S. E(S) means the class entropy of a subset S, namely

$$E\left(S\right)=-\sum_{i=1}^{k}P\left(C_i,S\right)\log P\left(C_i,S\right) \tag{8}$$

Where $P\left(C_i,S\right)$ is the proportion of examples in S that have class C_i and the logarithm may be converted to any convenient base.

We will calculate weighted normalized decision matrix with interval numbers as below:

$$\tilde{v}_{ij}^{l}=w_j\tilde{a}_{ij}^{l}, i=1,2,...,n, j=1,2,...,m \tag{9}$$

$$\tilde{v}_{ij}^{u}=w_j\tilde{a}_{ij}^{u}, i=1,2,...,n, j=1,2,...,m \tag{10}$$

where w_i is the weight of the i^{th} criterion and $\sum w_i=1$.

Fuzzy set is an extended form of classic set introduced by Zadeh. In a classic set, each element has two values. In other words, an element either belongs to a set or not. If an element becomes a member of set A, its related value is equal to 1, and zero, otherwise. However, fuzzy theory is attributing a number between [0, 1] to each x from X.

A Convex Fuzzy Set: The "A" fuzzy set is convex if and only if each $x_1,x_2\in X$ and each $\lambda\in[0,1]$, we have

$$\mu_A\left[\lambda x_1+(1-\lambda)x_2\right]\geq\min\left[\mu_A\left(x_1\right),\mu_A\left(x_2\right)\right] \tag{11}$$

The cut point T_A for which E(A, T_A, S) is minimal amongst all the candidate cut points is taken as the best cut point.

Like other machine learning methods - systems that learn from data - neural networks have been used to solve a wide variety of tasks that are hard to solve using ordinary rule-based programming, including computer vision and speech recognition. Computational devices have been created in CMOS, for both biophysical simulation and neuromorphic computing. More recent efforts show promise for creating Nano-devices for very large scale principal components analyses and convolution. If successful, these efforts could usher in a new era of neural computing that is a step beyond digital computing, because it depends on learning rather than programming and because it is fundamentally analogous rather than digital even though the first instantiations may in fact be with CMOS digital devices.

Decision tree generating algorithm is a method of forming the research study proposed in the late 1960s and early 1970s, by an American student John Holland and his colleagues at the University of Michigan. In this method, mechanism simulation of biological evolution Model to construct artificial system, has been widely used in recent years. The traditional BP neural network algorithm has shortcomings such as slow convergence and easy to fall into Local minima. This paper, based on decision tree generating algorithm BP neural network mechanisms, improved the convergence speed of the network, and then used the improved BP neural networks for evaluation of the level of the university library information.

Particular way is, firstly code the model's initial weight values and threshold, constitute the initial population, generate the next generation population with the decision tree generating algorithm, then decode the best individuals of the population we obtain the weight, threshold and make evaluation; If it meets the design requirements, then output the optimal weights and threshold, otherwise proceed to the operation of decision tree generating algorithm until get the best individual of a generation population based on decision tree generating algorithm, output weights and the threshold value, which is the weight and threshold value based on global optimum of network. Then they are assigned to the BP network for final training.

Decision tree generating algorithm is a search algorithm based on natural selection and genetic mechanisms for groups, and simulates the course of reproduction, and the interbreeding in the process of a natural and genetic selection. Using decision tree generating algorithm, every possible solution is coded as "a chromosome", namely the individual, a number of individuals form groups (all possible solutions). Decision tree generating algorithm is the process of evaluating a generational group which is the community of a feasible solution.

4. THE EXPERIMENT AND ANALYSIS FOR EXPERIMENTAL DATA

The development environment of distributed OLAP system is Microsoft Visual C++ 6.0. The data warehouse is established with SQL Server 2000, and OLAP servers are built through SQL Server 2000 Analysis Services. The operating environment of whole system is Windows 2000 Server. OLAP servers are connected with LAN and they communicate through TCP/IP protocol. Each server has the same data warehouse and OLAP analysis services and job scheduling procedure.

Fig. (**3**) shows the query performance comparison of single, double and three OLAP server systems with a single kind of task. Thin broken curve shows the performance result that all leaf data are queried through a single server. Thick broken curve is the performance characteristic curve with double servers and full curve is characteristic curve with three servers. We can know from Fig. (**3**), while the

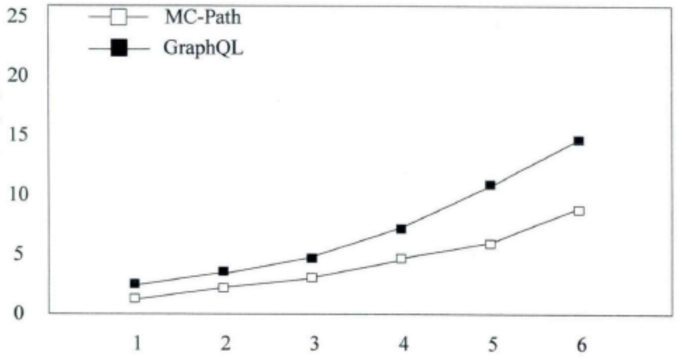

Fig. (3). The query performance of the system.

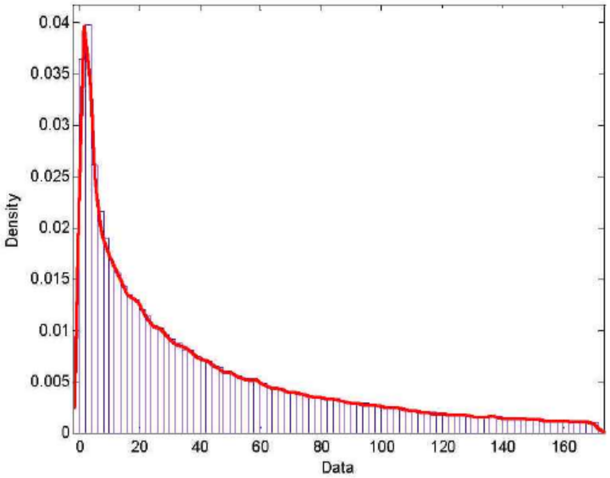

Fig. (4). The query performance of the system.

system has only one server; the query processing time increases almost linearly with the increase of the query objects' number. While the system has double or three servers, the relationship between processing time and the number of query objects is non-linear, it gives a smooth growth curve.

Compute the class information entropy of partitioning induced by the non-stable cut point (it is discussed in next section), select the cut point with minimal class entropy of partitioning and determine the best cut point. With the same increased number of query objects, the time increase speed of single server system is the fastest, second is the double servers system, and it is slowest for the three-server system. So we can know that with the server increase, the whole system's performance has been improving, but, the improvement of performance is nonlinear because of communication costs.

From the analysis of the system performance, we can draw the conclusion: the complexity of the new improved bully algorithm is O (n), and when the next bigger node initiates the selection process, there are least messages and its quality is n-2.

Through experiment, the selection processes are performed using the two algorithms in the distributed OLAP system that has 20 nodes. We can calculate the different messages quality through controlling the node that initiates the selection process.

In the Fig. (**4**), the full curve shows the performance result of bully algorithm and the broken curve shows that of the new improved one. According to the bully algorithm, the messages increase rapidly with the number of the initiating selection decreasing. And the new improved one can maintain low message quality, so the system is highly efficient and highly stable.

CONCLUSION

With the continuous development of decision support system, more and more enterprises begin to deploy OLAP application. But the OLAP system with one server is inefficient and expensive. The introduction distribute' technology into OLAP system can assign tasks to different servers. So the OLAP application can run on servers with low configuration or microcomputers. And the system can provide services to user as a whole with the help of the coordinator. In the distributed OLAP systems, the coordinator is selected according to selection algorithm. In this paper, we proposed a new improved bully algorithm. Through the experiments and analysis, we find that the algorithm can select the high performance coordinator and maintain low messages quality. When there are more nodes in the system, the superiority of the algorithm can be better embodied. In a less number of nodes system, it has the performance similar to the classic bully algorithm.

CONFLICT OF INTEREST

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

ACKNOWLEDGEMENTS

This work is supported by the Key Project of Guangxi Social Sciences, China (No.gxsk201424), the Education Science fund of the Education Department of Guangxi, China (No. 2014JGA268), and Guangxi Office for Education Sciences Planning, China (No.2013C108).

REFERENCES

[1] K.W. Chau, Y. Cao, and M. Anson, "Application of data warehouse and decision support system in construction management," *Automation in Construction*, vol. 3, pp. 213-224, 2003

[2] X. Li, B. He, and L. Zhong, "Bully algorithm and optimization in distributed systems," *Journal of Xi'an Technological University*, vol. 3, pp. 210-213, 2004.

[3] Z. Jin, and G. Hu, "Application of inquiry optimized in the Distributed database system," *Computer Applications and Software*, vol. 11, pp. 58-60, 2013.

[4] J. Han, and Q. Li, "A novel static task scheduling algorithm in distributed computing environments," In: *18th International Parallel and Distributed Processing Symposium*, pp. 3-12, 2004.

[5] P. Cominos, and N. Munron, "PID controllers: recent coming methods and design to specification," *IEEE Proceeding of Control Theory and Applications*, vol. 1, pp. 46-53, 2012.

[6] A. Lou, *The Essential Guide to Data Warehousing*, Prentice-Hall, 2000.

[7] V. Panos, and S. Timos, "A survey on logical models for OLAP databases," *ACM SIGMOD Record*, vol. 4, pp. 64-69, 2009.

[8] Z. Hou, "On analysis and processing of CRM based," In: *The 5th wuhan international conference on R-Business Data Mining and Data Warehouse whhan*, China, pp. 228-231, 2006.

Received: May 26, 2015 Revised: July 14, 2015 Accepted: August 10, 2015