International Journal of Advance Research in Science and Engineering
Volume No.07, Special Issue No. (02), January 2018
www.ijarse.com
IJARSE
ISSN: 2319-8354

Performance Evaluation of A Data Mining in the Cloud Storage

Hariom Rathore [1], Dr. Amit Sharma[2],

[1]M.Tech Scholar, [2]Professor, Computer Science & Engineering Department,
Vedant College of Engineering and Technology, Kota, Rajasthan (India)

ABSTRACT

Now a days cloud computing has become a cost effective and practical solution for data-intensive data mining technologies which produce highly sensitive, private and a trustful services. A series of cloud security controls are already in the market still concerned about the internal security loopholes viewed by data analyst. This paper focus on carried to analyze whether Information retrieval can improve in better way to show the performance. Mainly an Information retrieval protocol focuses on to those data mining application which specify a better security. Analyzing over multiple datasets with different sizes using tools the performance can be evaluated of Information retrieval and entire data mining applications. The Information retrieval is capable of encrypting the results of queries while producing the correct query results. Mainly focus to found the inefficient under the experimental data mining application with large dataset with indicating the use of big data and other encryption methods should also be investigated in order to secure data mining results faster.

Keywords: MINING, YARN

I. INTRODUCTION

Data mining is an important computer science technique for gather information and extract patterns and knowledge from large amount of data, used in games, business, human rights, medical, science and engineering with other fields. Because of costly hardware requirement organizations are least interested in data mining technique. But thanks to cloud computing environment the data mining technologies are adopted by various organizations in fewer amounts. But the problems of security and privacy always blink in the mind of person. Cloud computing solve security issues to an extent but still it is not convince to all.

Already it seen that lots of efforts are to be taken for delivering cloud computing services. Through which data mining techniques received with low cost, reliable, efficient and centralized format with implement management techniques. The information retrieval protocol is designed to protect user information from server side with suitable encryption protocol to use in such scenario. Still there were lots of scope viewing in respect to combining the data mining technologies, cloud computing and encryption to investigate performance. As we seen that the cloud computing with its resources is able to accelerate the process and information retrieval protocol securing the information but the performance of this combination is still remain unknown. So it require to combine it to decide which one is the valid option for protecting the data value in such environment. In this

process the setup is to established for checking performance, processing time and the adopting way into data mining system under cloud environment.

II. BACKGROUND

2.1 Performance Measurement Methodology

Today, server performance measurement has been studied extensively, but researches on the cloud computing performance measurement have just begun in the past few years [5]. The performance measurement methodology calculates the server performance value according to the test program execution time, CPU, memory, and other parameters after running a benchmark program on the server system [6]. For different uses of servers, its performance indicators are also different [7]. For example, when servers are mainly used for scientific computing, there is a high requirement for the computing speed of CPU; when servers are used for large-scale database processing [8], there is a high requirement for the memory capacity, access speed and the read and write speed of external memory.

The running time of benchmark program is the waiting time ranging from a certain standard task is input to the computer to the needed result is obtained, including time to access memory, CPU running time, disk I/O operation time and system operation time. In multi-tasking system, it would become more complex when CPU turns to other tasks while waiting for I/O operations. Therefore, when discussing performance, we sometimes use CPU time. It refers to CPU working time, which does not include I/O waiting time and running time for other tasks. Obviously, what the users see is the total amount of time spent when the program is over rather than only CPU time.

Nowadays, performance measurement methodology [9].

Mainly takes the following three factors into consideration:

1) Correctness : The correctness of the measurement results should always be put in the first place, as the purpose of running any test programs is to get the correct calculations;

2) Reliability: The operating system manages and controls the software and hardware resources of the server. When the system is running, users are able to make rational use of computer resources to run various programs; if the computer system is unstable, it will lead to abnormal hardware work and unstable software application.

3) Working capability: We focus on the processing capacity and responding capability. Processing capacity means the amount of information that can be processed per unit time. Responding capability includes response time, turnaround time and queuing time.

2.2 Hadoop Architecture

Hadoop is a distributed framework for processing the large data sets [21]. It is an Apache Lucene sub-project, which is maintained by the Apache Software Foundation. Its core components are HDFS and HDFS (Hadoop Distributed File System). And it is derived from Google's MapReduce and Google File System (GFS) papers. HDFS is an open source java product similar to GFS which is a distributed file system with high reliability and fault tolerance. MapReduce is a software framework for parallel processing of large data sets [10]. Through the

use of Hadoop HDFS and MapReduce technology, users can develop parallel applications in case they do not understand the underlying technology to handle large data sets.

1) HDFS

Hadoop Distributed File System is designed to be suitable for commodity hardware. HDFS is similar to the traditional hierarchical file system. It supports file read, write, delete, and other operations [11]. The starting point of establishing HDFS is based on high tolerance and low cost. Its framework is based on a master-slave architecture which is composed of a group of specific nodes. These nodes include one NameNode, which provides metadata service inside HDFS; DataNodes, which provide storage blocks for HDFS. Since there is only one NameNode, single point of failure exists in Hadoop. NameNode is the master which manages file namespace, file operations and Client access permissions. There is usually one DataNode module for each node in Hadoop cluster to manage the storage attached to the node. DataNodes also need to receive NameNode's command to process data blocks. Files stored in HDFS is divided into several blocks, all blocks are of the same size except for the last one. All blocks will be copied for tolerance. In order to improve the data tolerance, data blocks are copied to several DataNodes. The size of the block (usually 64MB) and the copied number can be configured. The architecture of HDFS is shown as Fig. 1.

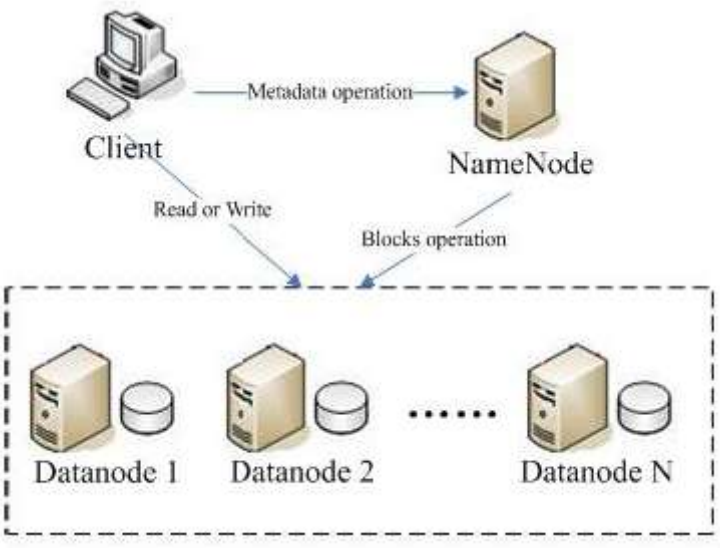

Figure 1. Architecture of HDFS

2) MapReduce

MapReduce is a software framework proposed by Google, which is used to parallel process large data set [12]. MapReduce architecture consists of one Job Tracker and a number of Task Trackers. The Job Tracker runs on the master node. It is responsible for scheduling job, managing task implementation, and communicating with Task Tracker. Task Tracker handling the tasks assigned by the Job Tracker is run on slave nodes. When user submits a Map Reduce program to Job Tracker, the job will be divided into several map tasks and reduce tasks. Then tasks are assigned to Task Trackers by Job Tracker [22]. Fig. 2 shows the architecture of Map Reduce.

International Journal of Advance Research in Science and Engineering
Volume No.07, Special Issue No. (02), January 2018
www.ijarse.com
IJARSE
ISSN: 2319-8354

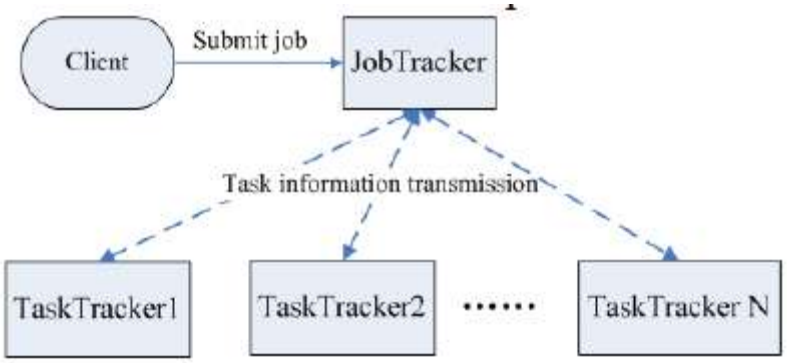

Figure 2. Architecture of Map Reduce

During the execution of a job, the main phases are the implementation of map and reduce tasks. In Map phase, the main work is reading data blocks and splitting into Map tasks in parallel processing. Each Map task possesses the input split and output the key-value intermediate result. The result is temporarily stored in the memory and disk. The work in reduce stage is concentrating the output of the same key to the same Reduce task and processing it, output the final result.

The advantage of Map Reduce lies in its tolerance and scalability. Its excellent scalability is the main factor that pushes the rising of its position, and its strong tolerance enables the high stability. In fact, MR is easy to understand, just as Google always advocates that the easiest problem-solving method is usually the most efficient one. Its theory can be summarized as follows: divide to-be-processed file into several parts, thereby divide the tasks, and then integrate the tasks to complete them [13]. It is a divide-integrate process, which can complete various kinds of tasks, but not all.

III. EXPERIMENT DESIGN AND METHODS

A three layered abstract framework consists of the philosophy layer, the technique layer and the application layer. The philosophy investigates the fundamental problems with data. The study of information mining is a precursor to technology and application. The technique layer is the study of information discovery ways and their implementation in machine. It can be logical or physical. To be achieved by logical analysis, mathematical modeling and programming language. The application layer is to effectively use the discovered knowledge to form express and precise the intuitive notions of utility and significance of discovered knowledge.

A significant implication of the framework lies on its division of the understanding of a posh downside into completely different levels that ends up in a division of basic problems with data processing into levels. It conjointly provides the right context within which a specific kind of data at the technique level.

Apache Mahout offers several types of algorithm with which users can analyses dataset including Collaborative filtering, Classification, Clustering and Dimensionally Reduction. With the help of Apache Mahot tool Maven it's easy to designed and managed any project view with any dataset.

There are several popular existing tests of groups of results such as one-sample test, two-sample test, paired test, t-test, F-test. T-test are used to compare means under relaxed conditions and determine whether two samples are significantly different from each other.

International Journal of Advance Research in Science and Engineering
Volume No.07, Special Issue No. (02), January 2018
www.ijarse.com
IJARSE
ISSN: 2319-8354

$$t = \frac{(\sum D)/N}{\sqrt{\frac{\sum D^2 - \left(\frac{(\sum D)^2}{N}\right)}{(N-1)(N)}}}$$

ΣD: Sum of the differences

ΣD^2: Sum of the squared differences

$(\Sigma D)^2$: Sum of the differences squared.

An application can be developed with the use of K-mean algorithm from apache Mahout under Hadoop environment. The data set to be uploaded on HDFS and Yarn is able to convert the dataset to mahout sequence files of vector Writable files. The Yarn application can be used as iteration stages and result including cluster point. The clustering results include node points, information retrieval processing time and total processing time will be generated and send back to HDFS. The information retrieval processing time and total processing time will be later will be analyzed by the evaluation methods.

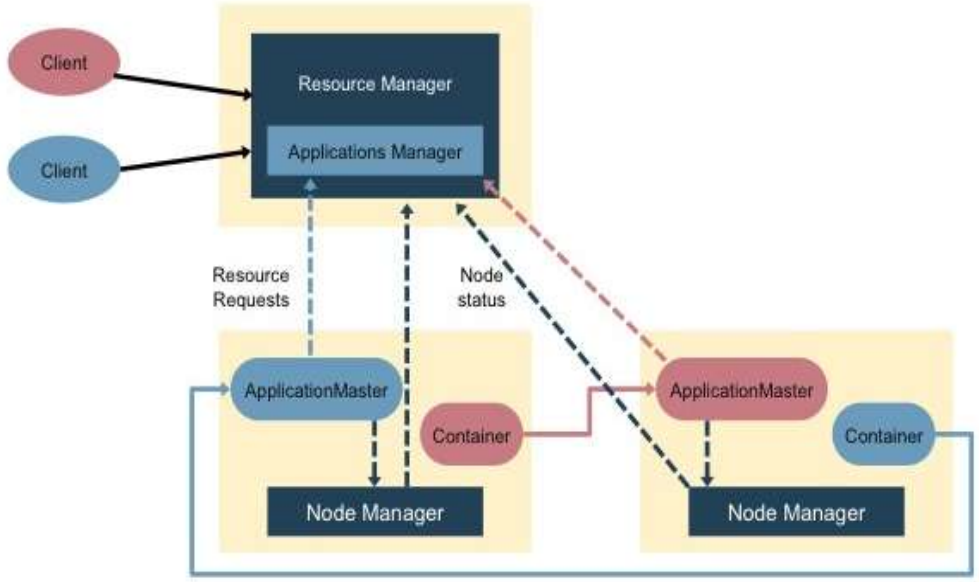

Figure 3 YARN MR Application Execution Flow

As per the basis of classification accuracy, number of unclassified instances and computational complexity it can be seen that decision tree is one of the best choice for cloud computing area. The classification accuracy and number of unclassified instances essentially summaries the average performances of the techniques.

IV. EXPERIMENT DESIGN AND RESULT

In this section, in order to demonstrate that the proposed method used to measure the nodes' performance in Hadoop, experiments on a real-world Hadoop cluster are conducted. In our experiment, we first select the appropriate performance benchmarks to measure node performance value, and then we run benchmark programs in Hadoop cluster. The benchmarks need to be run several times on Hadoop cluster, and the measurement results

are averaged. Finally, we verify the measurement of the node performance value in accordance with the measurement results.

4.1. Experimental Environment and Parameters

The experimental cluster has seven Servers which contains one Name Node and six Data Nodes. The server operating system is Centos 5.4, and the Hadoop version is 1.0.2. The hardware parameters of each Server are shown in Table I. As shown in Table I, the performance of servers is difference in this Hadoop heterogeneous cluster. For example, node 2 is IBM X236. From the aspect of hardware parameter, this machine is not much worse than the other machines, but it has been used six years and its performance has been decreased to some extent. So the hardware parameter of servers cannot be a good measurement of the server performance. We need to use performance benchmarks to get the fair and objective measurement result.

In previous sections, we have already made detailed introduction to server performance measurement, and clarified the focus of node performance measurement in Hadoop cluster, mainly including server CPU processing capability, memory performance, disk performance and network performance. In our experiment, Hadoop cluster is mainly used in data analysis. Therefore, we focus on the CPU and memory performance. In (1), the value of α and β should be greater. In this experiment, we choose Unix Bench as a measurement tool. Unix Bench is a performance benchmark with a long history, and its measurement results reflect the overall performance of a server. Theoretically, Unix Bench measurement results have a direct relationship with the CPU, memory, storage, operating systems. However, according to our observations, for modern computer systems, measurement results were more affected by the impact of the CPU processing capacity. Hence, we choose Unix Bench measurement results to represent the server's performance value. Meanwhile, after Hadoop jobs running experiment, according to the experimental results we can use the (5) to measure whether the performance measurement results are correct.

Word Count is a classic Hadoop test program, which is used to calculate the number of occurrences of each word in the specified data set. It is one of the basic algorithms for processing text data. During the testing process, Word Count program will process the specified data and output the calculation result in accordance with the number of occurrences of the words. So we will use Word Count as our test program.

4.2 Experiment Method and Results Analysis

In this experiment, we first start Hadoop daemon processes on these nodes. Then we use performance benchmark to measure nodes performance when the Hadoop cluster is idle. Next, we upload data into cluster, run the test program with different size of data sets, and get the test results. Finally, we analyze the test results to verify whether performance measurement results are correct.

We select the test procedures suitable for current application scenarios from Unix Bench. File Copy is used to measure the rate when data is transferred between different files. This test of file read, write and copy can obtain the number of characters which can be written, read and copied in a specified time. Process Creation is used to calculate the number of times that a process creates and harvests a child process which immediately exits. This test directly depends on the memory bandwidth. The Shell Scripts measures the number of times process can start and harvest a set of parallel copy shell scripts in one minute. Since the server's CPU is multicore in our

International Journal of Advance Research in Science and Engineering
Volume No.07, Special Issue No. (02), January 2018
www.ijarse.com
IJARSE
ISSN: 2319-8354

experimental environment, we select the number of parallel shell scripts is 8 and 16. System Call Overhead uses the system call time to calculate the cost of entering and leaving the operating system kernel. Then we use Unix Bench to multiple measure servers in Hadoop cluster, and the measurement results are averaged. The performance measurement result of nodes is shown in Fig 4.

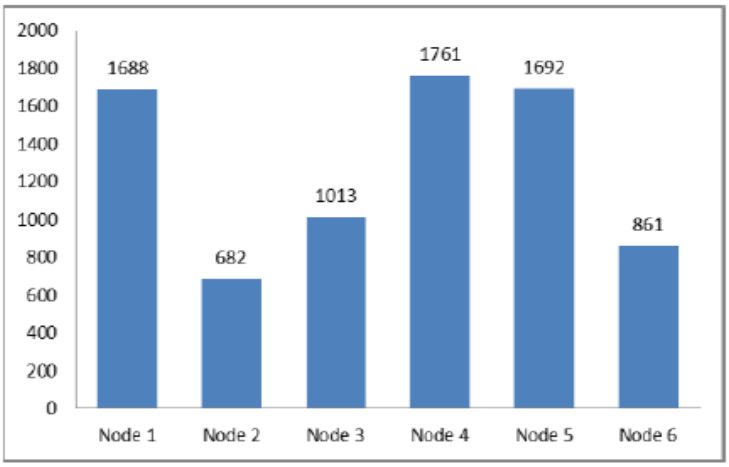

Figure 4. Performance value of each node

Using the performance measurement we can see the performance value of nodes, as shown in Fig. 4. In this figure, we can see that the performance difference between each node is larger. The reason is that server hardware parameters are difference; meanwhile, several servers have been used for a long time, which can lead to a decline in performance. This also shows the necessity of using performance benchmarks to measure server performance.

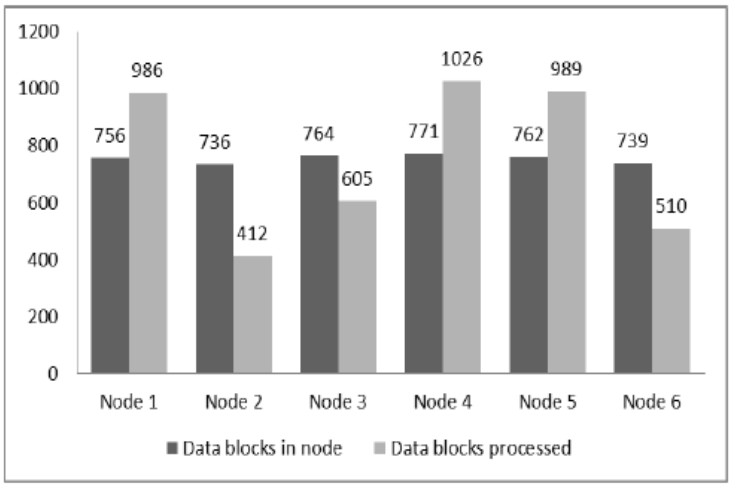

Figure 5. The distribution of the number of data blocks

V. CONCLUSION

Security issue still a most challenging topic in cloud computing. As per this paper internal security issue for those who providing data mining serviceman encryption method was proposed. In Information retrieval protocol system a comprehensive data mining tool, algorithm and cloud framework is evaluated and get a relevant result

for selection. According to the experiment results the information retrieval focus on correct information received. Evaluation result shows that the processing time increase rate of information retrieval and data mining system are similar. The result shows that for small set of data sets work efficiently in cloud.

REFERENCES

[1] Armbrust M, Fox A, Griffith R, et al. A view of cloud computing. Communications of the ACM. 2010, 53(4) pp. 50-58.

[2] White T. Hadoop pp. The definitive guide. O'Reilly Media, 2012.

[3] Shvachko K, Kuang H, Radia S, et al. The hadoop distributed file system. IEEE, 2010.

[4] Zaharia M, Konwinski A, Joseph A D, et al. Improving mapreduce performance in heterogeneous environments. USENIX Association, 2008.

[5] Steinmetz D, Perrault B W, Nordeen R, et al. Cloud Computing Performance Benchmarking and Virtual Machine Launch Time. ACM, 2012.

[6] Hatt N, Sivitz A, Kuperman B A. Benchmarking Operating Systems. 2009.

[7] Fadika Z, Dede E, Govindaraju M, et al. Benchmarking mapreduce implementations for application usage scenarios. IEEE, 2011.

[8] Gu Y, Grossman R. Toward Efficient and Simplified Distributed Data Intensive Computing. Parallel and Distributed Systems, IEEE Transactions on. 2011, 22(6) pp. 974-984.

[9] Franks R G. Performance analysis of distributed server systems. Carleton University, 1999.

[10] Borthakur D. HDFS architecture guide. HADOOP APACHE PROJECT http://hadoop. apache. org/common/ docs/ current/hdfs design. pdf. 2008.

[11] Borthakur D. The hadoop distributed file system pp. Architecture and design. Hadoop Project Website. 2007, 11 pp. 21.

[12] Vijayalakshmi V, Akila A, Nagadivya S. THE SURVEY ON MAPREDUCE. International Journal of Engineering Science. 2012, 4.

[13] Shim K. MapReduce algorithms for big data analysis. Proceedings of the VLDB Endowment. 2012, 5(12) pp. 2016-2017.

[14] Wang K L. The Performance Analysis of Cloud and Traditional Computing Architectures by Emulating Navy Ship Repair and Maintenance Information System. 2011.

[15] Vedam V, Vemulapati J. Demystifying Cloud Benchmarking Paradigm-An in Depth View. IEEE, 2012.

[16] Capps D, Norcott W D. IOzone filesystem benchmark. 2008.

[17] Staelin C. lmbench pp. an extensible micro- benchmark suite. Software: Practice and Experience. 2005, 35(11) pp. 1079-1105.

[18] Schroder C. Measure Network Performance with Iperf. February, 2007.

[19] Wu J, Chi H, Chi L. A Cloud Model-based Approach for Facial Expression Synthesis. Journal of Multimedia. 2011, 6(2) pp. 217-224.

[20] Lei Y, Lai H, Li Q. Geometric Features of 3D Face and Recognition of It by PCA. Journal of Multimedia. 2011, 6(2) pp. 207-216.