Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 2, Issue. 4, April 2013, pg.513 – 516

RESEARCH ARTICLE

IMPLEMENTATION OF PARALLEL APRIORI ALGORITHM ON HADOOP CLUSTER

A. Ezhilvathani[1], Dr. K. Raja[2]
[1]P.G Student, M.E CSE, Alpha College of Engg, Chennai, India
[2]Dean (Academics), Alpha College of Engg, Chennai, India

Abstract— Nowadays due to rapid growth of data in organizations, large scale data processing is a focal point of information technology. To deal with this advancement in data collection and storage technologies, designing and implementing large-scale parallel algorithm for Data mining is gaining more interest. In Data mining, association rule learning is a popular and well researched method for discovering interesting relations between variables in large databases. This paper aims to extract frequent patterns among set of items in the transaction databases or other repositories. Apriori algorithms have a great influence for finding frequent item sets using candidate generation. Apache Hadoop software framework is used to build the cluster. It working is based on MapReduce programming model. It is used to improve the processing of large-scale data on high performance cluster. It processes vast amount of data in parallel on large cluster of computer nodes. It provides reliable, scalable, distributed computing.

Key Terms: - Hadoop; MapReduce; Apriori

I. INTRODUCTION

Data mining can be defined as the process of discovering hidden pattern in database. The main aim of the data mining is to manipulate the data into knowledge. Association rule mining is a kind of data mining process. Association rule mining is done to extract interesting correlations, patterns, associations among items in the transaction database or other data repositories. Association rules are widely used in various areas such as telecommunication networks, marketing and risk management, and inventory control etc. In this paper Apriori algorithm is used to find the frequent item set in database. This is the method for finding the set of all possible combination of items and then counts the support for them. The parallel association rule mining can be categorized in two sections [5,9]. The first is data parallelism in which the input data set could be divided among the participating node to generate the rules. The second method is of dividing the task among the nodes so that each node will access the whole input data set for generating the rules.

Hadoop is a free, Java-based programming framework that supports the processing of large data sets in a distributed computing environment. It is part of the Apache project sponsored by the Apache Software Foundation. Hadoop was originally conceived on the basis of Google's MapReduce, in which an application is broken down into numerous small parts [10]. Hadoop can provide much needed robustness and scalability option to a distributed system as Hadoop provides inexpensive and reliable storage. The Apache Hadoop software library can detect and handle failures at the application layer, so it can deliver a highly-available service on top of a cluster of computers, each of which may be prone to failures.

II. RELATED WORKS AND EXISTING MODEL

The Nirali R, Sheth and J. S. Shah has implemented Association Rule based parallel data mining algorithm which deals with Hadoop cloud, a parallel store and computing platform [1].

In this paper Association Rule based parallel data mining algorithm which deals with Hadoop cloud, a parallel store and computing platform is defined Moreover cloud inter operation between Hadoop and Sector/Sphere Cloud which allows the same Hadoop MapReduce application to be run against data in either Hadoop Distributed File System or Sphere File System have been introduced. Nowadays huge amount of data are created every day. With this rapid explosion of data, processing is moved from terabytes era to petabytes era. This trend creates demand for advancement in data collection and storing technology. Hence there is a growing need to run data mining algorithm on very large datasets. Hadoop is the software framework for writing applications that rapidly process vast amounts of data in parallel on large clusters of compute nodes. It works on MapReduce programming model. MapReduce is a generic execution engine that parallelizes computation over a large cluster of machines.

MapReduce is a distributed Programming Model intended for large cluster of systems that can work in parallel on a large dataset. The Job Tracker is responsible for handling the Map and Reduce process. The tasks divided by the main application are firstly processed by the map tasks in a completely parallel manner. The MapReduce framework sorts the outputs of the maps, which are then input to the reduce tasks. Both the input and output of the job are stored in the file system. Due to parallel computing nature of MapReduce, parallelizing data mining algorithms using the MapReduce model has received significant attention from the research community since the introduction of the model by Google. The MapReduce model based on Hadoop is examined for applicability in the field of Data Mining.

Algorithm: F1= (Frequent itemsets of cardinality 1);
For (k=1; Fk$\neq\varphi$;k++)do begin
C_k+1=apriori-gen (F_k);//New candidates
For all transactions t$\in$Database do begin
C't=subset (C_k+1,t);//Candidates contained in t
For all candidate c$\in$C't do
c.count++;
End
F_k+1={C$\in C_k$+1 |c.count$\geq$minimum support}
End
End
Answer $\cup_k F_k$;

Apriori algorithm finds all frequent itemsets by scanning the database time after time [3]. This algorithm wastes a lot of time and memory space so to introduce parallelization in Apriori algorithm, an improved Apriori algorithm is proposed which is shown below.

(i).In first step for doing parallel scan first split the transaction database horizontally into 'n' data subsets and distribute it to 'm' nodes.

(ii).Then each node scans their sown data sets and generate set of Candidate itemset C_p.

(iii).Then the support count of each Candidate itemset is set to 1. This Candidate itemset C_p is divided into r partitions and sent to 'r' nodes with their support count.'r' nodes respectively accumulate the support count of the same itemset to produce the final practical support, and determine the frequent itemset L_p in the partition after comparing with the minimum support min_sup.

(iv). finally merge the output of 'r' nodes to generate set of global frequent itemset L.

The above improved Apriori algorithm is used to considerably reduce the time as in this algorithm getting frequent item sets by traversing transaction database only once. This performance degradation with Sector file system can be due to I/O overhead where there is a great deal of file transfer between the distributed and local file system. Another issue can be JNI overhead as it faces difficulty while going through JNI Layer to access Sector.

III. SYSTEM DESCRIPTION

In this paper, Hadoop software framework is used for processing vast amounts of data in parallel on single node cluster [4]. It works on MapReduce programming model. MapReduce is a generic execution engine that parallelizes computation over a large cluster of machines. Apriori algorithm finds all frequent itemsets by scanning the database time after time.

Implementation of single node Hadoop cluster: Hadoop is a framework written in Java for running applications on large clusters of commodity hardware and incorporates features similar to those of the Google File System and MapReduce [2]. HDFS is a highly fault-tolerant distributed file system and like Hadoop

designed to be deployed on low-cost hardware. It provides high throughput access to application data and is suitable for applications that have large data sets.

Hadoop Web Interfaces: Hadoop comes with several web interfaces which are by default (see conf/hadoop-default.xml) available at these locations:

http://localhost:50070/ – web UI of the NameNode daemon

http://localhost:50030/ – web UI of the JobTracker daemon

http://localhost:50060/ – web UI of the TaskTracker daemon.

These web interfaces provide concise information about what's happening in Hadoop cluster.

Implementation of wordcount example: WordCount is a simple application that counts the number of occurrences of each word in a given input set. WordCount example reads text files and counts how often words occur. The input is text files and the output is text files, each line of which contains a word and the count of how often it occurred, separated by tab.

Each mapper takes a line as input and breaks it into words. It then emits a key/value pair of the word. Each reducer sums the counts for each word and emits a single key/value with the word and sum.

An optimization, the reducer is also used as combiner on the map outputs. This reduces the amount of data send across the network by combining each word into a single record.

Assuming HADOOP_HOME is the root of the installation and HADOOP_VERSION is the Hadoop version installed, compile WordCount.java and create a jar:

Compile the program:

hduser@ubuntu:~$ javac -classpath ${HADOOP_HOME}/hadoop-core-1.0.3.jar -d wordcount_classes Desktop/WordCount.java

Create the jar :

hduser@ubuntu:~$ jar -cvf Desktop/wordcount.jar -C wordcount_classes/ .

To display input:

hduser@ubuntu:~$ /usr/local/hadoop/bin/hadoop dfs -cat /user/hduser/word/file0

hadoop hello world

bye world

Run the Application:

hduser@ubuntu:~$ /usr/local/hadoop/bin/hadoop jar Desktop/wordcount.jar WordCount /user/hduser/word /user/hduser/word_out

Implementation of Apriori algorithm for frequent itemset generation on hadoop: Apriori algorithm is implemented with MapReduce Programming model as shown below

(i).In first step Partitioning and Distributing data - Transaction database is divided into 'n' subsets by MapReduce Library and are sent to 'm' nodes executing Map tasks.

(ii).Insecond step Formatting the data subsets- Format 'n' data subsets as <key1,value1> pair where key is transaction id.

(iii).In third step Execute Map task - The task of Map function is to scan each record of the input item subset and generating Candidate itemset C_p.

(iv).In fourth step Operate Combiner option - Combiner function firstly combines the Mapfunction outputs in the local and outputs <itemset,support_count> . Combiner function then uses partition function to divide the intermediate pairs generated by combiner function into r different partitions.

(v).Finally Execute Reduce task at Reduce function, the key itemsets are first sorted. After sorting the Reduce function add up the support count of the same candidates and get the actual support count of the candidate in the whole transaction database. Then comparing it with the minimum support count and getting the frequent itemsets L_p.

IV. CONCLUSION

This paper has implemented a single node Hadoop cluster which is working based on MapReduce model. Using this Hadoop cluster, wordcount example is implemented. This paper also extracts frequent patterns among set of items in the transaction databases or other repositories using Apriori Algorithm in single node Hadoop cluster. It can be easily parallelized and easy to implement.

REFERENCES

[1] Nirali R. Sheth, J.S. Shah, May 2012 "Implementing Parallel Data Mining Algorithm on High Performance Data Cloud" ISSN: 2277 – 9043 International Journal of Advanced Research in Computer Science and Electronics Engineering Volume 1, Issue 3.

[2] Apache hadoop. http://hadoop.apache.org.

[3] "Top 10 algorithms in data mining", © Springer-Verlag London Limited 2007.
[4] Michael G. Noll, August 5, 2007,"Running Hadoop On Ubuntu Linux (Single-Node Cluster)".
[5] Agrawal R, Srikant R,1994, "Fast algorithms for mining association rules" Proceedings of the 20th VLDB conference, pp 487–499.
[6] R. Agrawal and J.C. Shafer, December 1996, " Parallel mining of association rules",IEEE Transactions on Knowledge and Data Eng., 8(6):962–969.
[7] Suraj P . Patil1, U. M. Patil2 and Sonali Borse3, World Journal of Science and Technology 2012,"The novel approach for improving apriori algorithm for mining associationRule",2(3):75-78,ISSN: 2231 – 2587.
[8] Han E , October 1999 "Text categorization using weight adjusted k-nearest neighbor classification", PhD thesis,University of Minnesota.
[9] Srikant R, Agrawal R, 1995,"Mining generalized association rules. In: Proceedings of the 21st VLDB conference. pp. 407–419.
[10] J. Dean, S. Ghemawat, Dec. 2004 "MapReduce: Simplified Data Processing on Large Clusters," In Proc. of the 6th Symposium on Operating Systems Design and Implementation, San Francisco CA.

