RESEARCH ARTICLE OPEN ACCESS

Analysis of Big Data Processing Using Data Mining

Shashank Dubey[1],Vidya Chitre[2]
Department of Information Technology, VIT,Mumbai,India

----------------------------------***********************-------------------------------------

Abstract:

Cloud computing is a powerful technology that are highly used to perform large- scale and complex computing. It completely remove requirement to maintain expensive computing hardware, or software and large space. Massive growth has been observed in the generation of data or big data over the year through cloud computing. For processing and anlaysing result from that large data generally require lots of computational power and large infrastructure requirement. Hence there is a need to conjoined big data with cloud computing. Since this big data is important to analysis in order to extract insight knowledge from the data. Traditional association rule mining for frequent itemset which scan the dataset into main memory may become inconvenient when handling large dataset. Hence there is need of processing this dataset on multiple commodity machine using parallel technique of Map Reduce Framework. This paper gives overview of different big data mining algorithm used for processing dataset.

Keywords —**Big Data,Hadoop,MapReduce,Data Mining**

----------------------------------***********************-------------------------------------

I. INTRODUCTION

In this modern era, there is sudden increase in amount of data generated and ability to collect this large data has increased significantly because of advances in hardware and software platforms. For example, Wal-Mart alone handles more than 1 million customer transactions each hour and imports those into databases estimated to contain more than 2.5 petabytes of data. Web log data sites such as Facebook and Twitter handles, stores and generated terabyte to petabyte of data each day alone and this number keeps on growing. Since the data is often so large that specialized methods are required for the mining process. With this extremely large data set it may be difficult or infeasible for single machine to process and find association pattern rules between the data set. Because the traditional algorithm has issues of scalability, memory and computation cost, stability and low performances when it comes to deal with this huge data. Also the streaming and big-data architectures are slightly different and pose different challenges for the mining process. When processing this big data for the problem of frequent itemset there is need to consider a lots of challenges. A major problem arises when the data is large enough to be stored in a distributed way. Therefore, significant costs are incurred in shuffling the data or the intermediate results of the mining process across the distributed nodes. These costs are also referred to as data transfer costs. Therefore when handling large dataset, then the algorithms need to designed to take into account both the disk access constraint and the data transfer costs. In addition, many distributed frameworks such as MapReduce require specialized algorithms for frequent pattern mining. The focus of big data framework is somewhat different from streams, in that it is closely related to the issue of shuffling large amounts of data around for the mining process. Interestingly, it is sometimes easier to process the algorithms in a single pass in streaming fashion, than when they have already been stored in distributed frameworks where access costs become a major issue.

Dealing with big datasets in the order of terabytes or even peta bytes is a challenging. Hence cloud computing provides an effective technique called Parallel programming which is becoming a necessity to deal with the massive amounts of data, which is produced and consumed more and more every day. Parallel programming architectures, and hence the algorithms, can be grouped into two major subcategories: shared memory and distributed (share nothing). On shared memory systems, all processing units can concurrently ac-cess a shared memory area.

On the other hand, distributed systems are composed of processors that have their own internal memories and communicate with each other by passing messages [10]. It is easier to adapt algorithms to shared memory parallelism in general, but they are typically not scalable enough [10].

II. LITERATURE SURVEY

Association rule mining was introduced by R.Agrawal in 1993. It still has an active research area in the data mining and machine learning. Association rule mining finds correla-tions between items in a database. The classic application for association rule mining is market basket analysis , in which business experts aim to investigate the shopping behaviour of customers in an attempt to find some commonness . The aim is to find groups of items that are frequently sold together in order that marketing experts can develop strategic decisions concerning shelving, sales promotions and planning. Associ-ation rule mining has been widely used in various industries beside supermarkets such as mail order, telemarketing, and e-commerce.

The author R.Agarwal[1] has proposed an Apriori which scan the database to generate candidate k-itemset and it has to repeatedly scan k+1 times for k itemset. To overcome this limitation, JW. Han, J. Pei and YW. Yin.[2] has proposed FP-Growth algorithm which scan the database twice and accordingly it construct the FP-tree. Based on FP-tree which

TABLE I

COMPARISON OF APRIORI, FP-GROWTH AND MFP ALGORITHM

Parameter	Apriori	FP-Growth	MFP
Techniques	It uses the apriori property and eliminates the items that does not satisfy the minimum count	It construct FP-tree from which conditional frequent itemset is generated that satisfy the minimum Count	It uses the TL table along with MFP-tree to generate frequent itemset
Utilization of Memory	It generate large candidate which may require large memory space	It does not require much memory space since it uses the FP-tree	It also does not require large memory space.
Number of Scans	It scan the database multiple times to generate candidate itemset	It scan the database only twice.	It only require once database scan.
Execution Times	Lots of times wasted for producing the candidate itemset	Execution times is less as compared to Apriori algorithm	Require less time than FP-Growth and Apriori algorihtm

TABLE II

SUMMARY ANALYSIS OF BIG DATA MINING TECHNIQUES

Sr No	Author	Title	Algorithms	Observation Remarks
1	Yang, Xin Yue, Zhen Liu, and Yan Fu	MapReduce as a programming model for association rules algorithm on Hadoop[4]	Parallel Apriori	It uses the Hadoop technology to improve the traditional Apriori Algorithm
2	Haoyuan Li,Yi Wang,Ming Zhang	PFP: Parallel FP-Growth for Query Recommendation[5]	Parallel FP- Growth	It consider the memory and computation cost of processing large dataset and hence it proposed the Parallel FP-Growth which partitions the data on commodity machine for processing
3	Moens, Sandy, Emin Aksehirli, and Bart Goethals	Frequent itemset mining for big-data[6]	Dist-Eclat and BigFIM	The author has proposed two technology to speed up and optimized the run on large scale data
4	Rong, Zhuobo, Dawen Xia, and Zili Zhang	Complex statistical analysis of big data: implementation and application of apriori and FP-growth algorithm based on MapReduce[7]	Apriori and FP-Growth based on MapReduce	The author has consider the problem of single machine environment for processing large dataset. Hence it implemented the traditional algorithm (Apriori and FP-Growth) on MapReduce parallel environment to overcome the issues of memory utilization,scalability and low performances
5	Wei, Xiaoting	Incremental FP-Growth mining strategy for dynamic thresholdvalue and database based on MapReduce[8]	Parallelized Incremental FP-Growth	This algorithm realizes the effective data mining when threshold value and database changes at the same time
6	Liao, Jinggui, Yuelong Zhao, and Saiqin Long	MRPrePost-A parallel algorithm adaptedfor mining big data[9]	MRPrePost	MRPrePost is more superior than PrePost and Parallel FP-Growth algorithm in terms of scalability,stability and performances
7	Yen-huiLiang and Shiow-yang Wu	Sequence-Growth: A Scabale and Effective Frequent Itemset Mining Algorithm for Big Data Based on MapReduce Framework[10]	Sequence-Growth	A lexicographical order is used to construct a tree which help to find all frequent itemset from dataset without exhaustive search

stored in memory, it process to find frequent itemset. But it create a large number of conditional tree to process the tree which is recurrsive process.Gao, Jun.[3] also the same algorithm which follow the FP-Growth algorithm, but it reduces the scan of a database by one time. MFP algorithm is efficient in terms of time execution as compared to FP-Growth and Apriori. It can be used on any input dataset where FP-Growth or Apriori are applied.Table 1. show the comparison of these 3 algorithm.

III. REVIEW OF BIG DATA MINING ALGORITHM AND MAPREDUCE

This section will focus on the recent work done on handling big data through MapReduce technology. Table II. shown above give summary analysis of different algorithm the author has proposed and its observation result.

A. Mapreduce On Hadoop Framework

MapReduce is data parallel and scalable programming model which introduce by Google. The MapThe MapReduce consist of two function i.e mapper and reducer. The mapper function work on the input data which is split among the mapper on multiple node. Each mapper take as key ,value pair as input to generate output intermediate result. Then on this intermediate result MapReduce perform the shuffling and sorting operation to give the combined output from all mapper as input to reducer. The reducer function will aggregate all the <key,value> pair to generate the final output. Mapreduce has the advantage it automatically handle the complicated issues like the load balancing, distribution of data on multiple node and fault tolerances. MapReduce can be served on low commodity hardware and it is scalable which is another alternative to expensive infrastructure. The input data for MapReduce is mainly stored in HDFS which provide high I/O bandwidth when running on cluster of multiple node. Hadoop has several component namely Na-menode, Datanode, Jobtracker and TaskTracker. Namenode is the heart of HDFS and it is master server which maintain the metadata regarding the input which split across the node. Jobtracker is also master server which request for metadata to Namenode so as to assign and schedule the task for TaskTracker. TaskTracker is mainly responsible for execution of mapper and reducer task.

IV. FUTURE WORK

In this section we have applied the idea to implement the MFP algorithm on MapReduce Framework. Since the Parallel FP-Growth algorithm based on MapReduce also need to scan the database twice and it also recursive procedure to construct the conditional tree. Hence using MFP algorithm based on MapReduce Framework, we can overcome the limitation of excessive scanning database which improve the efficiencies and scalable of Parallel FP-Growth and MFP algorithm. Fig. show flow of future work.

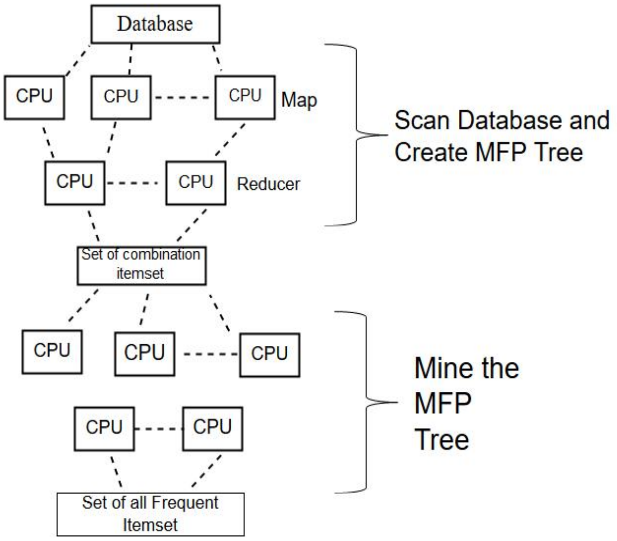
Database
CPU
CPU
CPU
Map
CPU
CPU
Reducer
Scan Database and Create MFP Tree
Set of combination itemset
CPU
CPU
CPU
CPU
CPU
Mine the MFP Tree
Set of all Frequent Itemset

Fig.1.Future Work Flow Chart

V. CONCLUSIONS

This paper gives the overview of different traditional and big data mining algorithm with its comparative analysis. We also suggested an new algorithm using MFP to implement based on MapReduce so as to improve the scalalbility and efficiencies problem.

REFERENCES

[1]R. Agrawal and R. Srikant.“Fast Algorithms for Mining Association Rules in Large Databases”,Journal of Computer Science and

Tech-nology, vol. 15, 1994, pp. 487-499.

[2] JW. Han, J. Pei and YW. Yin.“ Mining Frequent Patterns without Candidate Generation”, International Conference on Management of Data, vol. 29(2), 2000, pp. 1-12.

[3] Gao, Jun. “Realization of a new association rule mining algorithm.” cis. IEEE, 2007.

[4] Yang, Xin Yue, Zhen Liu, and Yan Fu.“MapReduce as a program-ming model for association rules algorithm on Hadoop.”Information Sciences and Interaction Sciences (ICIS), 2010 3rd International Conference on. IEEE, 2010

[5] Li, Haoyuan, et al. “Pfp: parallel fp-growth for query recommenda-tion.” Proceedings of the 2008 ACM conference on Recommender systems. ACM, 2008.

[6] Moens, Sandy, Emin Aksehirli, and Bart Goethals.“Frequent itemset mining for big data.” Big Data, 2013 IEEE International Conference on. IEEE, 2013.

[7] Rong, Zhuobo, Dawen Xia, and Zili Zhang. “Complex statistical anal-ysis of big data: implementation and application of apriori and FP-growth algorithm based on MapReduce.” Software Engineering and Service Science (ICSESS), 2013 4th IEEE International Conference on. IEEE, 2013.

[8] Sun, Li, et al. “An efficient algorithm for updating association rules with incremental transactions and minimum support changes simul-taneously.” Intelligent Systems (GCIS), 2012 Third Global Congress on. IEEE, 2012

[9] Liao, Jinggui, Yuelong Zhao, and Saiqin Long.“MRPrePost-A parallel algorithm adapted for mining big data.” Electronics, Computer and Applications, 2014 IEEE Workshop on. IEEE, 2014.

[10] Liang, Yen-hui, and Shiow-yang Wu.“Sequence-Growth: A Scalable and Effective Frequent Itemset Mining Algorithm for Big Data Based on MapReduce Framework.” 2015 IEEE International Congress on Big Data. IEEE, 2015.