Data Partitioning in Frequent Itemset Mining on Hadoop Clusters

Shivani Deshpande, Harshita Pawar, Amruta Chandras, Amol Langhe

[1]*Student, Dept. of JSPM's RSSOER, Pune, Maharashtra, India*
[2]*Student, Dept. of JSPM's RSSOER, Pune, Maharashtra, India*
[3]*Student, Dept. of JSPM's RSSOER, Pune, Maharashtra, India*
[4]*Student, Dept. of JSPM's RSSOER, Pune, Maharashtra, India*

---***---

Abstract - *For mining frequent Itemsets parallel traditional algorithms are used. Existing parallel Frequent Itemsets mining algorithm partition the data equally among the nodes. These parallel Frequent Itemsets mining algorithms have high communication and mining overheads. We resolve this problem by using data partitioning strategy. It is based on Hadoop. The core of Apache Hadoop consists of a storage part, called as Hadoop Distributed File System (HDFS), and a processing part called Map Reduce. Hadoop divides files into large blocks. It distributes them across nodes in a cluster. By using this strategy the performance of existing parallel frequent-pattern increases.*

***Key Words*: Frequent Item set Mining, Parallel Data Mining, Data Partitioning, MapReduce Programming Model, Hadoop Cluster.**

1. INTRODUCTION

Parallel Frequent Itemset mining is looking for sequence of actions and load balancing of dataset. Creating Hadoop cluster is especially for storage and analyzing data. Through frequent Itemset mining extracting knowledge from data. Example of this technique is Market Basket Algorithm. It also affect on load balancing. It helps to increase the speed of performance. This parallel Frequent Itemset mining is done using map reduce programming model. Partitioning of data in dataset through algorithm making data more efficient. This data partitioning is carried out on Hadoop clusters. Data partitioning necessary for scalability and high efficiency in cluster. In Frequent Itemsets Mining data partition affects to computing nodes and the traffic in network. Data partition may be spread over multiple nodes, and users at the node can perform local transactions on the partition. This increases performance for sites that have regular transactions involving certain views of data, whilst maintaining availability and security. By using Fidoop-DP concept, performance of parallel Frequent Itemset Mining on Hadoop clusters increases.Fidoop-DP is voronoi diagram. It is conceptualized on data partition strategy.

1.1 Software and Hardware Requirement

1.1 Software Requirement

- Nodes
- RAM
- Storage
- Network

1.2 Hardware Requirement

- Hadoop Cluster (Map Reduce)
- Windows Operating System
- Data Sets

2. Literature Survey:

Yaling Xun, Jifu Zhang, Xiao Qin, "FiDoop-DP: Data Partitioning in Frequent Itemset Mining on Hadoop Clusters", 2016.It describes, A data partitioning approach called FiDoop-DP using the Map Reduce programming model. The overarching goal of FiDoop-DP is to boost the performance. A similarity metric to facilitate data-aware partitioning. As a future research direction, we will apply this metric to investigate advanced load balancing strategies on a heterogeneous Hadoop cluster.

I.Pramudiono &amp;M.Kitsuregwa," Fp-tax: Tree structure based generalized association rule mining",2004.This paper describes, Investigation of data partioning issues in parallel FIM.Main focus is on map-reduce. Future work is development of Fidoop which exploits correlation among traction to partition large datasets in Hadoop.

X.Lin," Mr-apriori:Association rules algorithm based on mapreduce",2014.It explain, Main focus on classical Algorithm connecting and pruning step using prefix Itemset based storage using has table. It points some limitations of Apriori algorithm.

S. Hong, Z.Huaxuan, C. Shiping, and H.Chunyan," The study of improved fp-growth algorithm in mapreduce",2013.This describes, Build cloud platform to implement the parallel FP-growth algorithm based on linked list and PLFPG.PLFPG algorithm compared higher efficiency and scalability.

M. Liroz-Gistau, R. Akbarinia, D. Agrawal, E. Pacitti, and P. Valduriez," Data partitioning for minimizing transferred data in mapreduce",2013.It state that, Map Reduce jobs are executed over distributed system composed of a master and set of workers. Input is dividing into several splits and assigned to map tasks. Future work is evasion to perform the repartitioning in parallel

3. Existing System:

The following are technologies which has used in Existing Method.

- Map-Reduce: It is a processing technique and a program model for distributed computing based on java.
- Data Partitioning in Hadoop Clusters: It controls the Execution of parallelism of Hadoop clusters.
- FP-growth algorithm: It is used to find Frequent itemsets.
- Hadoop: It is used to develop applications that could perform complete statistical analysis on huge amounts of data.

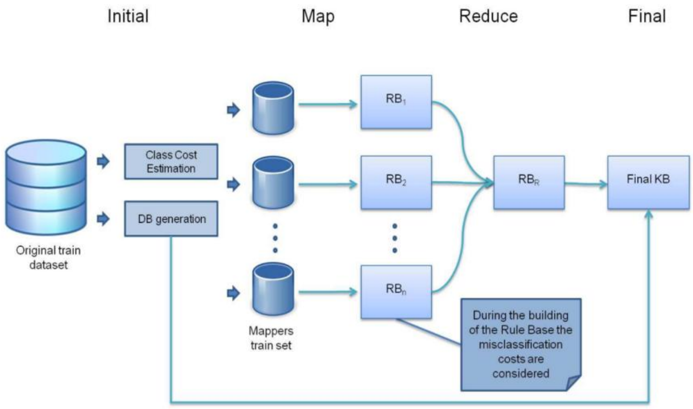

Fig-1: MapReduce Model

Map-Reduce is a promising parallel and scalable programming model for data-intensive applications and scientific analysis. A Map-Reduce program expresses a large distributed computation as a sequence of parallel operations on datasets of key/value pairs. A Map-Reduce computation has two phases namely, the Map and Reduce phases. The Hadoop runtime system establishes two processes called Job Tracker and Task Tracker. JobTracker is responsible for assigning and scheduling tasks; each TaskTracker handles Map or Reduce tasks assigned by JobTracker.

A partition is a division of a logical database or its constituent elements into distinct independent parts. Database partitioning is normally done for manageability, performance or availability reasons, as for load balancing. Hadoop Map Reduce determines when the job starts how many partitions it will divide the data into.

The FP-Growth Algorithm, proposed by Han in, is an efficient and scalable method for mining the complete set of frequent patterns by pattern fragment growth, using an extended prefix-tree structure for storing compressed and crucial information about frequent patterns named frequent-pattern tree (FP-tree).

3. Proposed system:

This section is about objective of project, which tells detail of system. It reduces the complexity of data access and retrieval. When we have to dealing with big data i.e. huge amount of data traditional exisisting system seems inefficient. The alternative to this is apache Hadoop, which deals with big data with efficiency. Hadoop itself consists of Map Reduce and HDFS.We use Hadoop with concept called frequent Itemsets which makes it FiDoop(Frequent item Hadoop).It runs on Hadoop cluster

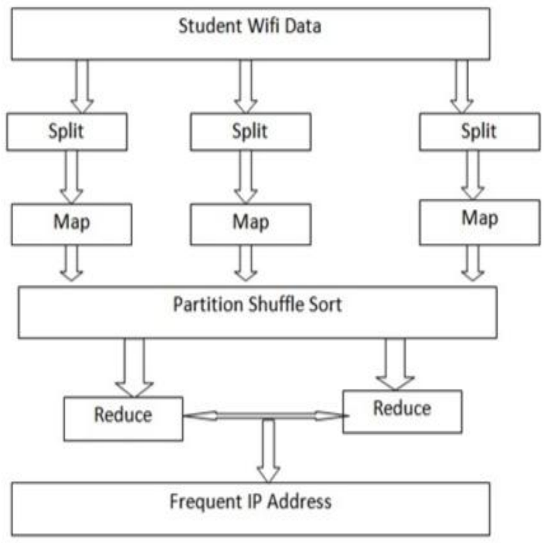

Fig-2: System Architecture

Job of map reduce is partitioning of data, it splits the local input data to generate local 1-itemset and it further reduce to specific reduced data. It sorts the data in decreasing order of frequency.

In second step job of map reduce is FP-Growth based partitions and last job is last task to aggregate the result from previous stages to generate output.

LSH based partitioning boost the performance of system by avoiding large number of comparisons. It uses bucket to keep similar transaction together.

4. CONCLUSIONS

To deals with migration of high communication, and reduce computing cost in map reduce. We use frequent item data partitioning which establishes correlation among transaction for data partitioning.

It is based on:

- Similarity in data and transaction;
- Group this highly correlated data.

Group this highly correlated datacontent comes here Conclusion content comes here . Conclusion content comes here The existing references tell that frequent item mining improves the output up to 31% with18% average.

We are working to develop system that investigates the detail of students. [Uses Wi-Fi].It allows generating result based on various parameters.

REFERENCES

[1] Yaling Xun, Jifu Zhang, Xiao Qin,FiDoop-Dp Data Partitioning in Frequent Itemset Mining on Hadoop clusters,2016.

[2] S. Sakr, A. Liu, and A. G. Fayoumi, âœThe family of mapreduce and large-scale data processing systems, ACM Computing Surveys (CSUR), vol. 46, no. 1, p. 11, 2013.

[3] X. Lin, Mr-apriori: Association rules algorithm based on mapreduce,â in Software Engineering and Service Science (ICSESS), 2014 5th IEEE International Conference on. IEEE, 2014, pp. 141"144.

[4] S. Hong, Z. Huaxuan, C. Shiping, and H. Chunyan, âœThe study of improved fp-growth algorithm in mapreduce, in 1st International Workshop on Cloud Computing and Information Security. Atlantis Press, 2013.

[5] P. Uthayopas and N. Benjamas, Impact of i/o and execution scheduling strategies on large scale parallel data mining, Journal of Next Generation Information Technology (JNIT), vol. 5, no. 1, p. 78, 2014.

[6] Y. Xun, J. Zhang, and X. Qin, Fidoop: Parallel mining of frequent itemsets using mapreduce, IEEE Transactions on Systems, Man, and Cybernetics: Systems, doi: 10.1109/TSMC.2015.2437327, 2015.

[7] W. Lu, Y. Shen, S. Chen, and B. C. Ooi, Efficient processing of k nearest neighbor joins using mapreduce,â Proceedings of the VLDB Endowment, vol. 5, no. 10, pp. 1016â"1027, 2012.

[8] J. Leskovec, A. Rajaraman, and J. D. Ullman, Mining of massive datasets. Cambridge University Press, 2014.

[9] B. Bahmani, A. Goel, and R. Shinde, Efficient distributed locality sensitive hashing,â in Proceedings of the 21st ACM international conference on Information and knowledge management. ACM, 2012, pp.2174â“2178.

[10] M. Liroz-Gistau, R. Akbarinia, D. Agrawal, E. Pacitti, and P. Valduriez, âœData partitioning for minimizing transferred data in mapre- duce,â in Data Management in Cloud, Grid and P2P Systems. Springer,2013, pp. 1â“12.

BIOGRAPHIES

Ms. Shivani Deshpande
BE Computer Engg.

Ms. Harshita Pawar
BE Computer Engg.

Ms. Amruta Chandras
BE Computer Engg.

Mr. Amol Langhe
BE Computer Engg.