ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

A Survey on Parallel Mining of Frequent Itemsets in MapReduce

Shreedevi C Patil

M. Tech student, Dept of CSE, UBDT College of Engineering, Davanagere, Karnataka, India

ABSTRACT: Data mining faces a lot of challenges in the big data era. Association rule mining algorithm is not sufficient to process large data sets. Apriori algorithm has limitations like the high I/O load and low performance. The FP-Growth algorithm also has certain limitations like less internal memory. Mining the frequent itemset in the dynamic scenarios is a challenging task. A parallelized approach using the mapreduce framework is also used to process large data sets .The most efficient the recent method is the FiDoop using ultrametric tree (FIUT) and Mapreduce programming model. FIUT scans the database only twice. FIUT has four advantages. First: I reduces the I/O overhead as it scans the database only twice. Second: only frequent itemsets in each transaction are inserted as nodes for compressed storage. Third: FIU is improved way to partition database, which significantly reduces the search space. Fourth: frequent itemsets are generated by checking only leaves of tree rather than traversing entire tree, which reduces the computing time
.

KEYWORDS: MapReduce, frequent itemsets, ultrametric tree, Data mining, Apriori algorithm

I.INTRODUCTION

Data mining faces a lot of challenges in this big data era. The term big data refers to the voluminous amount of data which is difficult to store, analyze and process. The Big data includes various technologies to obtain useful information from the huge amount of data. The mining of big data is a difficult process. One of the main challenge of the data mining is the finding the frequent itemset.

Data mining is a process of discovering the pattern from the huge amount of data. There are many data mining technics like clustering, classification and association rule. The most popular one is the association rule that is divided into two parts i) generating the frequent itemset ii) generating association rule from all itemsets. Frequent itemset mining (FIM) is the core problem in the association rue mining. Sequential FIM algorithm suffers from performance deterioration when it operated on huge amount of data on a single machine to address this problem parallel FIM algorithms were proposed.

There are two types of algorithms that can be used for mining the frequent itemsets first method is the candidate itemset generation approach and without candidate itemset generation algorithm. The example for candidate itemset generation approach is the Apriori algorithm and for, without candidate itemsets generation is the FP-growth algorithm.

The important data-mining problem is discovering the association rule between the frequent itemset.in order to find best method for mining in parallel, we explore a spectrum for trade-off between computation, synchronization, communication, memory usage. Count distribution, data distribution, candidate distribution are three algorithms for discovering the associate rule between frequent itemsets.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

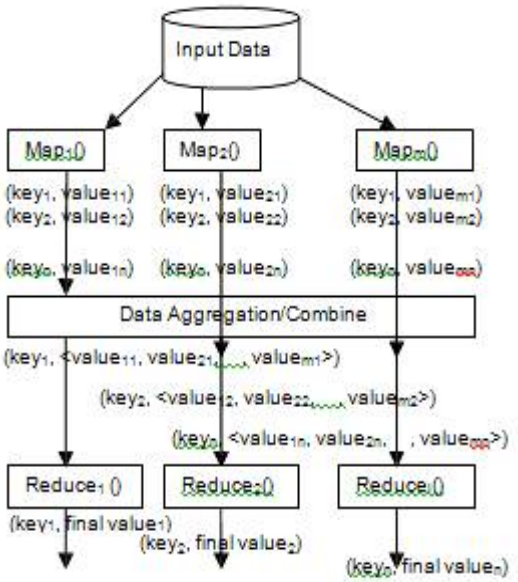

Fig. 1: Map or Reduce Flows

II. RELATED WORK

2.1 Mining of Frequent Itemsets

The *Apriori* algorithm is a classic way of mining frequent itemsets in a database [3]. A variety of *Apriori*-like algorithmsaim to shorten database scanning time by reducing candidate itemsets. For example, Park *et al.* [20] proposed the direct hashing and pruning algorithm to control the number of candidatetwo-itemsets and prune the database size using a hash technique. In the inverted hashing and pruning algorithm [21], every *k*-itemset within each transaction is hashed into a hash table. Berzal *et al.* [22] designed the tree-based association rule algorithm, which employs an effective data-tree structure to store all itemsets to reduce the time required for scanning databases. To improve the performance of *Apriori*-like algorithms, Han *et al.* [4] proposed a novel approach called FP-growth to avoid generating an excessive number of candidate itemsets. The main idea of FP-growth is projecting database into a compact data structure, and then using the divide-and-conquer method to extract frequent itemsets. The main bottlenecks of FP-growth are: 1) the construction of a large number of conditional FP trees residing in the main memory and 2) the recursive traverse of FP trees. To address this problem, Tsay *et al.* [13] proposed a new method called FIUT, which relies on frequent items ultrametric trees to avoid recursively traversing FP trees. Zhang *et al.* [23] proposed a concept of constrained frequent pattern trees to substantially improve the efficiency of mining association rules.

2.2 Parallel Mining of Frequent Itemsets

Parallel frequent itemsets mining algorithms based on *Apriori* can be classified into two camps, namely, count distribution (e.g., count distribution (CD) [6], fast parallel mining [24], and parallel data mining (PDM) [25]) and data distribution (e.g., data distribution (DD) [6] and intelligent data distribution [26]). In the count distribution camp, each processor of a parallel system calculates the local support counts of all candidate itemsets. Then, all processors compute the total support counts of the candidates by exchanging the local support counts. The CD and PDM algorithms have simple communication patterns, because in every iteration each processor requires only one round of communication. In the data distribution camp, each processor only keeps the support counts of a subset of all candidates. Each processor is responsible for sending its local database partition to all the other processors to compute support counts. In general, DD has higher communication overhead than CD, because shipping transaction data demands more communication bandwidth than sending support counts.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

2.3 Parallel Data Mining on Clusters

Clusters and other shared-nothing multiprocessor systems are scalable computing platforms that address the aforementioned main memory issue raised in parallel mining of large-scale databases. For example, Pramudiono and Kitsuregawa [27] proposed a parallel FP-growth algorithm running on a cluster. Javed and Khokhar [11] developed an efficient parallel algorithm using message passing interface on a shared-nothing multiprocessor system. Their PFP-tree-based parallel algorithm minimizes synchronization overheads by efficiently partitioning FP tree and the frequent-element list over processors. Tang and Turkia [28] used the extended conditional databases and *k*-prefix search space partitioning to parallelize FIM, and an implementation of the new scheme with FP trees is presented. Yu and Zhou [29] proposed two parallel mining algorithms, Tidset-based parallel FP-tree (TPFP-tree) and balanced Tidset-based parallel FP-tree (BTP-tree). The TPFPtree algorithm uses a transaction identification set to directly select transactions rather than scanning an entire database with a vertical database layout. Like most parallel mining algorithms running on clusters, TPFP-tree was implemented using the message passing interface programming model.

III. IMPLEMENTATION OF PARALLEL– APRIORI

In this section, we are going to redesign the Apriori algorithm to work parallel using the MapReduce. Map/Reduce paradigm, a clearly parallel system which can take full advantage of machine. This new design improves the efficiency of sequential Apriori in many aspects. Here we are using Hadoop environment, with the machine confirmation is CPU Pentium I-7, Ram 4.0 GB, Hard Disk 1.0 TB.

The input data set is then divided into the parts using classification techniques according to the items. Each of which is then assign to the Map function. Each Map function then process, localize input data to find out candidate 1 item set in form of <Key, Value > pair. Here the Key is individual item which is present in input dataset and Value is nothing but the number of occurrences of it. After computation of this, output of each Map function is passes to the data aggregation layer. Data aggregation layer combines all the data according to the key and generate global <Key, Value> pairs. In this stage, all the intermediate results are stored in the temporary file.

Later data stored on temporary file again split and passes to the Reduce function. Reduce function prunes the Key items which don't satisfy the minimum support criteria which are previously mention by the user. As per the property of an Apriori Algorithm, all nonempty subset of a non-frequent itemsetmust also be a non-frequent.The output of this stage is further provided as an input to the next iteration. This algorithm stops when there is no output file is present. The parallel Apriori helps in reducing the size of candidate itemsetas, it removes those itemsets whose subset were absent in the previous iteration's output file. Once the frequent itemsets are generated, association rules are developed.

Data Flow Diagram:

The proposed system design of DFD is shown below for overall representation of the Fidoop system under fig, the system allows the data modulation and demodulation approaches I understanding and technically perfuming the mining operations under the given scheme of evaluation. Initially the system is authenticated and then followed by the system behavioral approach of understanding and designing a extraction of datasets. These datasets are medical datasets under active chronic infection. In order to achieve an analytical result, the system has to be developed and designed in maintaining and segregating the system behavioral approach.

On extraction, the system retrieves the datasets from the input stream and the processing is initiated. Under this model of principle the system is aligned to perform the mining operation on big data under chronic infection and syndrome analysis.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

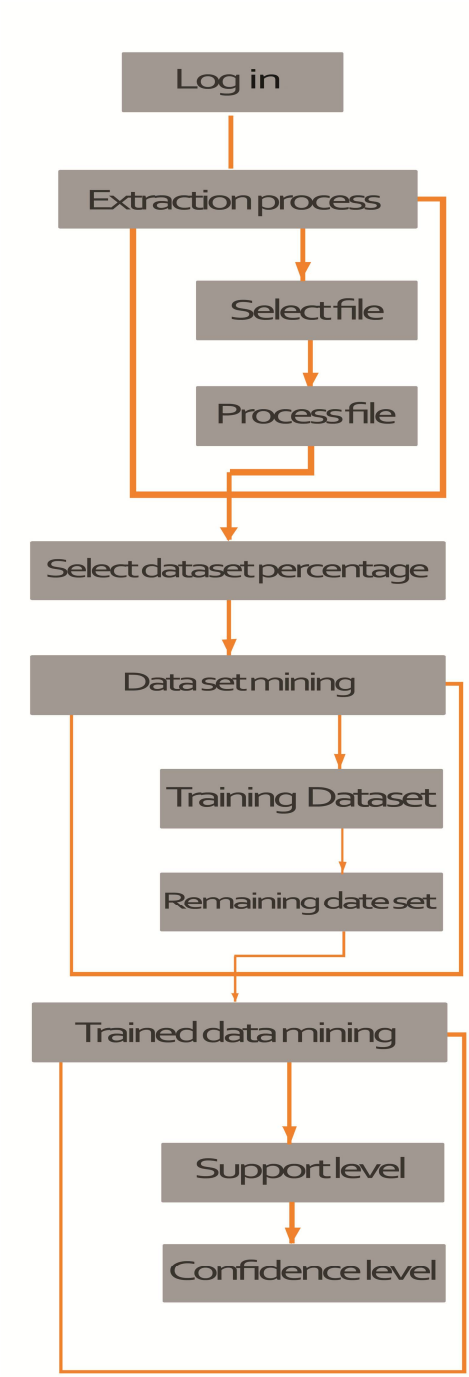

Fig. 2: Data Flow Diagram

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

IV CONCLUSIONS AND FUTURE WORK

Fidoop system has been dedicated to produce an accurate data mining results under Hadoop single node cluster environment, the system is simulated under Ubuntu for easy and high expert assistance. The proposed system under implementation shall produce appropriate results of support and confidence graph, the system support graph represents the high scale availability of the system under a random operating range and high confidence is been projected under confidence graph.

The system achieves high efficiency gain for providing static information resources for dynamic and critical data under big data mining. Results are detailed and discussed in previous chapters with overall system design and analysis. This system in future can be enhanced with a diplomatic sentiment anslysis and redefine process of computation under big data environment.

Apparently the proposed project can be used and appended in medical static data analysis and also the medical crisis and resource sharing analysis. This application is highly simulative and is active on all the medical conditions and diseases v/s resource mapping and decision analysis can be fetched.

REFERENCES

[1] “Parallel Mining of Association rule.” Rakesh Agarwal ,John C Safer.
[2] “Frequent Itemset Mining for Big Data Sandy Moens, Emin Aksehirli and Bart Goethals Universiteit Antwerpen, Belgium .
[3] “ECLAT Algorithm for Frequent Itemsets Generation “Manjit kaur , Urvashi Grag Computer Science and Technology, Lovely Professional University Phagwara, Punjab, India . International Journal of Computer Systems (ISSN: 2394-1065), Volume 01– Issue 03, December, 2014 Available at http://www.ijcsonline.com/ .
[4] “Implementation Of Parallel Apriori Algorithm On Hadoop Cluster” A. Ezhilvathani1, Dr. K. Raja. International Journal of Computer Science and Mobile Computing.
[5] “Frequent Itemsets Parallel Mining Algorithms “ Suraj Ghadge, Pravin Durge, Vishal Bhosale,Sumit Mishra. Department of Computer Engineering, JSPM’s ICOER. International Engineering Research Journal (IERJ) Volume 1 Issue 8 Page 599-604, 2015, ISSN 2395-1621.
[6] “FiDoop: Parallel Mining of Frequent Itemsets Using MapReduce” Yaling Xun, Jifu Zhang, and Xiao Qin, Senior Member, IEEE.
[7]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.
[8] Karim M, Hossain M, Rashid M, Jeong BS, Choi HJ. A MapReduce Framework for Mining Maximal Contiguous Frequent Patterns in Large DNA Sequence Datasets. IETETech Rev 2012;29:162-8.
[9] Han Jiawei, KamberMiceline. Fan Ming, MengXiaofeng translation, “Data mining concepts and technologies”. Beijing: Machinery Industry Press. 2001.
[10] R. Agrawal and J.C. Shafer , "Parallel Mining of Association Rules,"IEEE Tran. Knowledge and Data Eng. , vol. 8, no. 6, 1996,pp.962-96.
[11] Zhuobo Rong Sch. of Comput. & Inf. Sci., Southwest Univ., Chongqing, China Dawen Xia ; Zili Zhang “Complex statistical analysis of big data: Implementation and application of Apriori and FP-Growth algorithm based on MapReduce”.
[12] MapReduce Tutorial http://pages.cs.wisc.edu/~gibson/mapReduceTutorial.html.

BIOGRAPHY

Shreedevi C Patil is a MTech post graduate student in computer science and engineering, UBDT college of engineering, Davanagere ,Karnataka, India. She received her Bachelor of Engineering in computer science from Smt.Kamala and Sri.Venkappa M.Agadi College Of Engineering and Technology Laxmeshwar ,India in 2014.