ISSN: 2350-0328

International Journal of Advanced Research in Science, Engineering and Technology

Vol. 3, Issue 3 , March 2016

Efficient Algorithm for Frequent Itemset Generation in Big Data

Anbumalar Smilin V, Siddique Ibrahim S.P, Dr.M.Sivabalakrishnan

P.G. Student, Department of Computer Science and Engineering, Kumaraguru College of Technology, Coimbatore,India

Assistant Professor, Department of Computer Science and Engineering, Kumaraguru College of Technology, Coimbatore, India.

Associate Professor, Department of Computing Science and Engineering, VIT University, Chennai, India.

ABSTRACT: Data mining faces a lot of challenges in the big data era. Association rule mining is an important area of research in the field of data mining. 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. To overcome these issues a parallelized approach using the mapreduce framework has been used. The mining algorithm has been implemented using the Hadoop.

KEYWORDS: Incremental FP-growth algorithm, Bigdata, Data mining, Frequent itemset mining.

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. The data mining and the big data has lead to the emergence of the business intelligence. The data is gathered and analyzed to improve the profit of the organizations by adopting many techniques

The big data has certain unique characteristics like the volume, velocity, variety, veracity and value [8]. The big data has applications in many fields like the healthcare, governmental organizations, manufacturing industries, media, retail banking and research. The big data requires many technologies to obtain useful information. Big data has become the important area of research today. With the use of the social media like the facebook, twitter and many other social platforms the data is growing in a rapid manner. The fast growing data becomes the basis for the big data.

The association rule mining is an important area of research. The association rule mining becomes a tedious process in the case of the big data. The algorithms used for obtaining the frequent itemsets are not efficient in the case of the dynamic scenarios. The databases are updated dynamically. The threshold values like the minimum support count which are used for mining purposes are also updated dynamically. In such cases the mining process becomes a challenging task. The mining process has to be repeated every time the values are updated in the databases or the threshold values are reset. To overcome these issues a parallelized algorithm has been proposed.

A. DATA MINING

The main goal of the data mining is to extract useful information from the large datasets. Many hidden patterns are obtained from the data sets. The data mining involves many tasks like the anomaly detection, Association rule learning, clustering, classification, regression and summarization.

B. ASSOCIATION RULE MINING

It is an important data mining model used to find the interesting relationship between the data in the database. It is mainly used for the Market basket analysis to help improve the business activity. Association rules are obtained using two main criteria the support and the confidence. The support indicates how frequently the items appear in the database. The confidence refers to the number of times the if/then statements have found to be true.

C. FREQUENT ITEMSET MINING

It is mainly used for market basket analysis. The regularities in the shopping behavior of the customers can be found using the frequent itemset mining. The products which are bought together can be found using the frequent itemset mining.

D. DYNAMIC THRESHOLD VALUE

The threshold value refers to the values like the minimum support count which changes dynamically in the incremental databases. The minimum support count is used for obtaining the frequent itemsets. The threshold values play an important role in the pattern mining.

E. FP-GROWTH ALGORITHM

FP-Growth algorithm is mainly used to find the frequent itemset without candidate itemset generation. Two steps are followed in the FP-growth algorithm. In the first step, the FP-tree is constructed. In the second step the frequent itemset are extracted from the FP-tree.

II. LITERATURE REVIEW

JW.Han, J. Pei and YW.Yin [1] have proposed a new method called as the Frequent Pattern tree method. The frequent pattern tree stores the compressed information in an extended prefix tree structure. The frequent patterns are stored in a compressed form. A FP-tree based mining method known as the FP-growth is developed. The proposed algorithm helps in mining the frequent itemsets without the candidate set generation. Three techniques were employed to achieve the efficiency of mining. A large database is converted into a small data structure to avoid the repeated database scans which is said to be costly. It adopts a pattern frequent growth method to avoid generating large candidate sets which is very costly. The mining tasks are divided into smaller task which is very useful in reducing the search space. The FP-tree based mining also has many research issues like the SQL-based FP-tree structure with high scalability, mining frequent patterns with constraints and using FP-tree structure for mining sequential patterns.

H. Li, Y. Wang, D. Zhang, M. Zhang, and E. Chang [2] have proposed a parallel FP-Growth algorithm. In parallel FP-growth algorithm the mining task is divided into a number of partitions. Each of the partitions is provided to the different machines and each partition is computed independently. To overcome the challenges faced by the FP-growth algorithm like the storage, distribution of computation and highly expensive computation parallel FP-growth algorithm is proposed. The PFP algorithm consists of five steps. In the first step, the database is divided into small parts. In the second step the mapper and the reducer are used to do the parallel counting. In the third step the frequent items are grouped. In the Fourth step the FP-tree is constructed and the frequent itemsets are mined. In the fifth step the local frequent itemsets are aggregated. The PFP algorithm is effective in mining tag-tag associations and Webpage Webpage associations which are used in query recommendation or any other search.

Zhigang Zhang, Genlin Ji, Mengmeng Tang [3] has proposed a parallel algorithm MREclat based on Map/Reduce framework. In the vertical layout algorithm the frequent patterns are mined using the algorithm Eclat. The algorithms for mining frequent patterns in horizontal layout databases are different from the algorithms for mining vertical databases like the Eclat. A parallel algorithm MREclat which uses a mapreduce framework has been proposed to obtain the frequent itemsets from the massive datasets. Algorithm MREclat consists of three steps. In the initial step, all frequent 2-itemsets and their tid-lists are obtained from transaction database. The second step is the balanced group step, where frequent 1-itemsets are partitioned into groups. The third step is the parallel mining step, where the data got in the first step is redistributed to different computing nodes. Each node runs an improved Eclat to mine frequent itemsets. Finally, MREclat collects all the output from each computing node and formats the final result.MREclat uses the improved Eclat to process data with the same prefix. It has been proved that MREclat has high scalability and good speedup ratio.

Hui Chen, Tsau Young Lin, Zhibing Zhan and Jie Zhong [4] have proposed a parallel algorithm for mining frequent pattern in large transactional data. It uses an extended MapReduce Framework. A number of subfiles are obtained by splitting the mass data file. The bitmap computation is performed on each subfile to obtain the frequent patterns. The frequent pattern of the overall mass data file is obtained by integrating the results of all subfiles. A statistic analysis method is used to prune the insignificant patterns when processing each subfile. It has been proved that the method is scalable and efficient in mining frequent patterns in big data.
Xinhao Zhou and Yongfeng Huang [5] have proposed an improved parallel Apriori algorithm. The mapreduce framework is used to find the count and the candidate set generation. The proposed algorithm is compared with the existing traditional apriori algorithm. It has been proved that the proposed algorithm is more efficient compared to the traditional algorithm.

Jinggui Liao, Yuelong Zhao and Saiqin Long [6] have proposed a MRPrePost algorithm.It is a parallel algorithm which is implemented using the Hadoop platform. The MRPrePost is an improved PrePost algorithm which uses the mapreduce framework. The MRPrePost algorithm is used to find the association rules by mining the large datasets. The MRPrePost algorithm has three steps. In the first step the database is divided into the data blocks called the shards which are allocated to each worker node. In the second step the FP-tree is constructed. In the final step the FP-tree is mined to obtain the frequent itemsets. Experimental results have proved that the MRPrePost algorithm is the fastest.

Sheela Gole and Bharat Tidke [7] have proposed a new method, ClustBigFIM. Large datasets are mined using the Mapreduce framework in the proposed algorithm. BigFIM algorithm is modified to obtain the ClustBigFIM algorithm. ClustBigFIM algorithm provides scalability and speed which are used to obtain useful information from large datasets. The useful information can be used to make better decisions in the business activity. The proposed ClustBigFIM algorithm has four main steps. In the first step the proposed algorithm uses K-means algorithm to generate the clusters. In the second step the frequent itemsets are mined from the clusters. By constructing the prefix tree the global TID list are obtained. The subtrees of the prefix tree are mined to obtain the frequent itemsets. The proposed ClustBigFIM algorithm is proved to be more efficient compared to the BigFIM algorithm.

Surendar Natarajan and Sountharrajan Sehar [9] have proposed a new algorithm named Association rule mining based on Hadoop (ARMH). The proposed algorithm utilizes the clusters effectively and helps in mining frequent pattern from large databases. The workload among the clusters is managed using the hadoop distributed framework. The hadoop distributed file system stores the large database. Three mapreduce jobs have been used to mine the frequent patterns. The proposed ARMH algorithm obtains the frequent pattern from large databases.

III.METHODOLOGY

The original database is referred to as the D. The new datasets which are inserted into the database at a later stage is referred to as the D’. The original minimum support count is referred as T which is the threshold value. The updated minimum support count is referred to as the T’. The whole process is divided into the two phases.

In the First phase the frequent itemsets are obtained from the database D under the threshold value T. The first phase consists of two mapreduce tasks. The database D is divided into small chunks using a procedure called as the inputsplit. These chunks are sent to the mapper and the reducer to obtain the support count values. Using the support count value and based on the minimum support count the frequent list is formed. The frequent list formed is divided into groups.

The group list formed is sent to the mapper and the reducer to form the local FP-trees. From the local FP-trees the local frequent itemsets are extracted. The local frequent itemsets are integrated to obtain the frequent itemsets of the database D.

In the second phase, the new datasets are inserted into the database. The minimum support count is changed as T’. Based on the new support count T’ the frequent list is updated. The new updated grouplist is obtained. The new updated grouplist is given to the mapper and the reducer to obtain the local FP-trees. From the updated local FP-trees the local frequent itemsets are obtained. The local frequent itemsets are integrated to obtain the final result.

IV. ARCHITECTURAL DIAGRAM

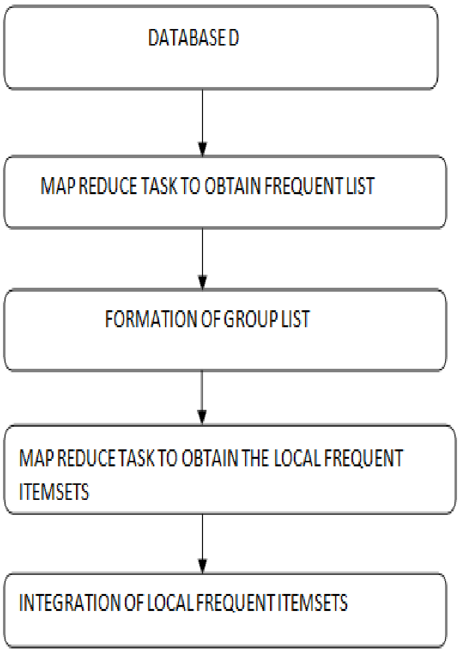

DATABASE D
MAP REDUCE TASK TO OBTAIN FREQUENT LIST
FORMATION OF GROUP LIST
MAP REDUCE TASK TO OBTAIN THE LOCAL FREQUENT ITEMSETS
INTEGRATION OF LOCAL FREQUENT ITEMSETS

Fig.1. Phase 1 Flow Diagram

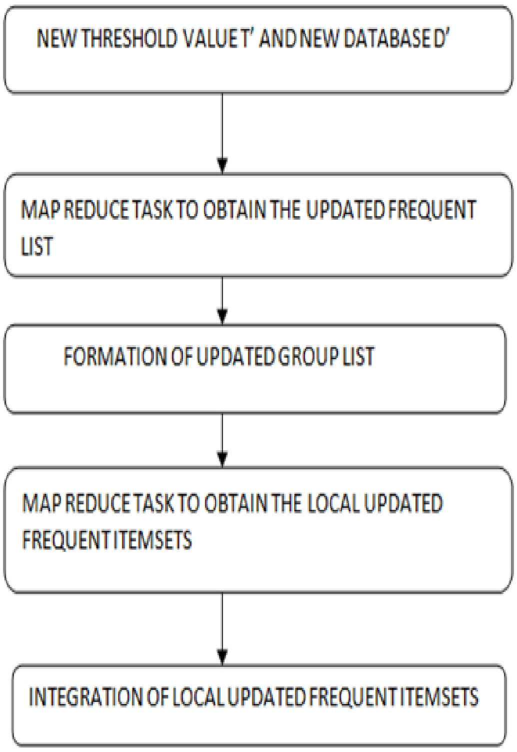

NEW THRESHOLD VALUE T' AND NEW DATABASE D'
MAP REDUCE TASK TO OBTAIN THE UPDATED FREQUENT LIST
FORMATION OF UPDATED GROUP LIST
MAP REDUCE TASK TO OBTAIN THE LOCAL UPDATED FREQUENT ITEMSETS
INTEGRATION OF LOCAL UPDATED FREQUENT ITEMSETS

Fig.2. Phase 2 Flow Diagram

V. ALGORITHM FOR THE PROPOSED TECHNIQUE

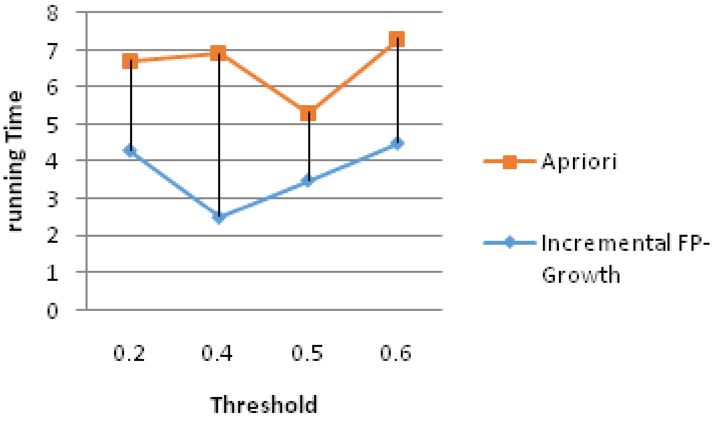
PROCEDURE PIFP-GROWTH (FP-TREE, F, T', T, D', D)

```
   $F_{1} = F1'-F_{1}$; // new frequent items of D and T'
   if ($F_{1!}$ = null)
      Update group list;
   Update frequentlist' //scan D'
   $F_{DP}$; // get powerful itemsets of D and T'
   $F_{D'P1}$, $F_{DP1;}$ // powerful frequent itemsets of D' and T, and D and T
   if ($F_{1}$'! = null)
       Update group list;
  if ($F_{1}$! = null and $F_{1}$'! = null)//update local FP-tree
      FP-tree' = conFPtree($F_{1}$ ', FP-tree,D')
  else
  {
  $F_{DD'}$=F;
  FP-tree'=FP-tree;
  Return;
  }
  Procedure conFP-tree ($F_{x,}$ tree, database) //update FP-tree
  For each t in group {//transaction t
      $F_{f}$' = $F_{f}$ + $F_{x}$;
Sort ($F_{f'}$');
nNode = t +$F_{x}$;
   insert nNode into tree; //insert new nodes
}
```

ISSN: 2350-0328
International Journal of Advanced Research in Science, Engineering and Technology
Vol. 3, Issue 3 , March 2016

VI.RESULT

The novel incremental FP-growth mining algorithm is implemented on the hadoop framework. The classical dataset T10I4D100K is used to implement the proposed algorithm. The T10I4D100K dataset consist of about 100,000 transactions and about 870 items. The proposed algorithm is found to cost the least amount of time when compared with the apriori algorithm. The proposed algorithm is found to be more optimized.



VII.CONCLUSION

Traditional association rule mining algorithm is not efficient in mining the big data. The existing algorithms cannot be applied in the dynamic scenarios. When the database is updated periodically and when the threshold values changes the mining process becomes a tedious task. These issues are overcome by the proposed novel incremental FP-growth algorithm which is implemented using the hadoop. So the proposed algorithm is found to be more effective.

REFERENCES

[1]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.

[2] H. Li, Y. Wang, D. Zhang, M. Zhang, and E. Chang, PFP: Parallel FP-growth for query recommendation, Proceedings of the 2008 ACM Conference on Recommender Systems, 2008, pp. 107-114.

[3] Zhigang Zhang, Genlin Ji, Mengmeng Tang, MREclat: an algorithm for parallel mining frequent itemsets, 2013 International Conference on Advanced Cloud and Big Data.

[4] Hui Chen, Tsau Young Lin, Zhibing Zhang and Jie Zhong, Parallel mining frequent patterns over big transactional data in extended mapReduce", 2013 IEEE International Conference on Granular Computing.

[5] Xinhao Zhou, Yongfeng Huang, An improved parallel association rules algorithm based on mapreduce framework for big data, 2014 11th International Conference on Fuzzy Systems and Knowledge Discovery.

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

[7] Sheela Gole, Bharat Tidke, Frequent Itemset Mining for Big Data in social media using ClustBigFIM algorithm, International Conference on Pervasive Computing.

[8] Siddique Ibrahim S P, Extract data in large database with hadoop, International Journal of Advances in Engineering and Scientific Research", Vol.1 2014, pp. 5-9.

[9] Surendar Natarajan, Sountharrajan Sehar, Distributed FP-ARMH algorithm in hadoop map reduce framework, 2013 IEEE.