COMPARATIVE STUDY OF DISTRIBUTED FREQUENT PATTERN MINING ALGORITHMS FOR BIG SALES DATA

Dinesh J. Prajapati
Research Scholar, Department of Computer Science & Engineering, Institute of Technology, Nirma University, Ahmedabad, India

ABSTRACT

Association rule mining plays an important role in decision support system. Nowadays in the era of internet, various online marketing sites and social networking sites are generating enormous amount of structural/semi-structural data in the form of sales data, tweets, emails, web pages and so on. This online generated data is too large that it becomes very complex to process and analyze it using traditional systems which consumes more time. This paper overcomes the main memory bottleneck in single computing system. There are two major goals of this paper. In this paper, big sales dataset of AMUL dairy is preprocessed using Hadoop Map Reduce that convert it into the transactional dataset. Then, after removing the null transactions; distributed frequent pattern mining algorithm MR-DARM (Map Reduce based Distributed Association Rule Mining) is used to find most frequent item set. Finally, strong association rules are generated from frequent item sets. The paper also compares the time efficiency of MR-DARM algorithm with existing Count Distributed Algorithm (CDA) and Fast Distributed Mining (FDM) distributed frequent pattern mining algorithms. The compared algorithms are presented together with experimental results that lead to the final conclusions.

Key words: Association rule, distributed frequent pattern mining, hadoop, map reduces.

Cite this Article: Dinesh J. Prajapati, Comparative Study of Distributed Frequent Pattern Mining Algorithms for Big Sales Data. International Journal of Advanced Research in Engineering and Technology, 8(1), 2017, pp 78–85.

1. INTRODUCTION

The process of data mining is to extract the useful information and patterns for the knowledge discovery process. One of the techniques used in data mining is called association rule mining. Association rule mining is the data mining task of uncovering relationships in the data. It is a popular model in the retail sales industry where a company is interested in identifying items that are frequently purchased together. An association rule is expressed in the form \( X \rightarrow Y \), where \( X \) and \( Y \) are the itemsets. This rule exposes the relationship between the itemset \( X \) with the itemset \( Y \). The interestingness of the rule \( X \rightarrow Y \) is measured by the support and confidence [1, 2]. The rule \( X \rightarrow Y \) has minimum support value \( \text{min}_\text{sup} \) if \( \text{min}_\text{sup} \) percent of transactions support \( XUY \), the rule \( X \rightarrow Y \) holds with minimum confidence value \( \text{min}_\text{conf} \) if \( \text{min}_\text{conf} \) percent of transactions which support \( X \) also support \( Y \) [3, 4]. Association rule mining process basically consists of two steps: (i) Finding all the frequent itemsets that satisfies minimum support
thresholds and, (ii) Generating strong association rules from derived frequent itemsets. Big data is termed for a collection of large data sets which are complex and difficult to process using traditional data processing tools [5].

In brief, the contribution of this paper is summarized in three steps: i) First of all, the distributed frequent itemset mining algorithms CDA, FDM and MR-DARM are used to generate the complete set of frequent itemsets and results are compared, (iv) Proposed framework mines not only frequent itemsets, but also mines distributor’s sales association rules in transactional datasets to analyze total sales based on the distributor. (v) Finally, based on user defined thresholds, the complete set of distributor’s sales strong association rules are generated with the interesting patterns. The CDA, FDM and MR-DARM distributed frequent mining algorithms are tested on sales dataset of AMUL Dairy.

The remaining of the paper is organized as follows. Related work is given in section 2. Section 3 shows the proposed methodology. In Section 4, the performance of CDA, FDM and MR-DARM algorithms are evaluated on sales dataset of AMUL dairy. Finally, the conclusion and future scope is drawn in section 5.

2. RELATED WORK

Authors in [6] proposed performance analysis factors like heterogeneous and autonomous. The authors also proposed a complex theorem which characterizes the features of both the big data revolution and big data processing model. Authors analyze the challenging issues in the data mining model and also in the big data analysis. Authors in [7] proposed imminent about big data mining infrastructures and analysis of Twitter. In this paper two major topics are discussed. First, schemas are insufficient to provide the knowledge of understanding the petabytes or terabytes of data. Second, a major challenge for analyzing the data is the heterogeneity of the various components. The objective of this paper is to share experiences of authors to analyze the data from Twitter in the area of production environment. Authors in [8] proposed an optimized distributed association rule mining approach to reduce the communication cost for geographically distributed data. The communication as well as computation time is considered to achieve an improved response time. The performance analysis is done based on scalability of processors in distributed environment. Authors in [9] proposed distributed trie based algorithm (DTFIM) to find frequent item sets. In this paper, authors proposed Bodon’s algorithm based on no shared memory in distributed computing environment. The proposed algorithm is revised with some frequent data mining algorithm. Authors in [10] proposed a distributed system for mining the transactional datasets using an improved Map Reduce framework. In this paper, authors implemented “Associated-Correlated-Independent” algorithm to find the complete set of customer’s purchase patterns along with the correlated, associated, associated-correlated, and independent purchase patterns.

The PARMA algorithm proposed in [11] provides great improvements to the runtime of finding association rules. PARMA achieves this by utilizing probabilistic results, it only approximates the answers. Another statistical approach was presented in [12]. This solution uses clustering to create groups of transactions and chooses candidate sets from the representative item sets in the clusters. Authors in [13] present improved version of the frequent item set mining algorithm as well as its generalized version. The authors introduced optimized formulas for generating valid candidates by reducing number of invalid candidates. By using the computations of previous steps by other processed nodes, it avoids generating redundant candidates. Authors also suggested to run the same algorithm in parallel or distributed system. The Count Distribution Algorithm (CDA) [14] provides fundamental distributed association rule algorithm. In this paper, each node contains huge number of frequent item sets and counts candidate item set locally. These count values are stored in the local database and maintains incoming count values. All the computing nodes execute the Apriori algorithm locally and after reading count values from the local database they broadcast respective count values to the remaining nodes. Each of the nodes can generate new candidate itemset based on the global counter. The FDM (Fast Distributed Mining) algorithm [15] provides candidate set generation algorithm similar to Apriori. The interesting property of local as well as global frequent itemset is used to generate a reduced set of candidates for the each iteration. Thus the
Comparative Study of Distributed Frequent Pattern Mining Algorithms for Big Sales Data

number of messages interchanged between each node reduces. Once the candidate sets are generated, then local reduction and global reduction techniques are applied to eliminate few candidate sets from each site.

In big data analysis, mining long patterns is more important for the transactional database having unique item set. However, none of the above mentioned work deals with the problem of data transformation and elimination of null transactions using Map Reduce. Therefore, data transformation and finding null transactions and then eliminating it for the future consideration; is the initial part of this proposed methodology. After removing null transactions, distributed frequent mining algorithm is applied to generate useful patterns. Existing CDA and FDM algorithm generates large candidate set, uses more number of message passing system and execution time is also higher while mining big data. The MR-DARM algorithm improves the drawback of CDA and FDM algorithms and generates useful patterns. The objective of this work is to remove the drawbacks of relational database and facilitate the existing Map Reduce framework; to generate the complete set of frequent itemsets with smaller candidate set generations, less message passing and improvement in the execution time of the system.

3. PROPOSED METHODOLOGY
The CDA and FDM algorithms are data parallelism algorithm [15]. In CDA algorithm, the dataset is divided into n number of partitions, each partition is given to separate node. Each node counts the candidates and then broadcasts its counts to remaining nodes. Each node then determines the global counts. The global counts are used to determine the large item sets and to generate the candidates for the next iteration. In FDM algorithm, candidate set is generated similar to Apriori algorithm. To reduce the size of candidates at each iteration, local and global frequent item sets are used which result reduction in the number of messages interchanged between nodes. Once the candidate sets are generated, local reduction and global reduction techniques are applied on each site to eliminate redundant candidate sets. The main drawback of CDA and FDM algorithm is that both generate large candidate set, uses more number of message passing system and execution time is higher while mining big data. These drawbacks can be improved by Map Reduce so the new approach is developed.

The MR-DARM algorithm is used to find frequent item sets from the actual transactional dataset. Once the actual transactional dataset is stored in HDFS, the entire dataset is split into the smaller segments and then each segment is transformed to data nodes. The map function is executed on each data segments and it produces <key, value> pairs for each record of database. The Map Reduce framework groups all <key, value> pairs, which have the same items and call the reducer function by passing value list for generating candidate item sets. In each database scan, map function generates local candidate item sets, then the reduce function generates global counts by adding local count values. For the overall computation, multiple iterations of Map Reduce functions are necessary. Each of the Map Reduce iteration produces a frequent item set. The iteration continues until no further frequent item sets are found. The reduce function adds up all the values produce by Mapper and generates a count for the candidate item. The main advantage of this approach is that it doesn’t exchange data between each node, but it only exchanges the count values. The MR-DARM algorithm uses notation $C_k$ as a set of candidate k-item set and $L_k$ as a set of frequent k-itemset which is shown in Fig. 1.
Input: Transactional Database in HDFS \((D)\), Minimum Support Threshold \((min\_sup)\)

Output: Frequent Itemsets \((L)\)

Method:

\(L_1 = \text{find frequent 1-itemsets from } D.\)

For each frequent \(k\)-itemset do

\[C_k = L_{k-1} \bowtie L_{k-1}. \quad \text{// Generates candidate itemset} \]

\[C_i = \text{Map}(). \quad \text{// Generates itemset occurrence} \]

\[L_k = \text{Reduce}(). \quad \text{// Gets the subset of frequent itemsets} \]

\[L = L_1 \cup L_k. \]

**Map Function:**

Input: Set of Transaction \((T_i)\)

Output: \(<\text{Candidate Itemset}, \text{Value}>\)

Method:

For each transaction \(T_i \in D\) do

For each itemset \(I_i\) in Candidate Itemset \(C_k\) do

If \((I_i \in T_i)\) then

Generate the output \(<I_i, 1>\)

as \(<\text{Key, Value}>\) pair.

**Reduce Function:**

Input: \(<\text{candidate itemset, list}>\)

Output: \(<\text{frequent itemset, support}\_\text{count}>\)

Method:

\(count = 0.\)

For each number in \(list\) do

\(count += \text{number}.\)

If \((count \geq Min\_sup)\) then

Generate the output \(<\text{frequent itemset, count}>\)

as \(<\text{key, value}>\) pair.

**Figure 1** The MR-DARM Algorithm

The transactional data is given as an input to the Mapper line by line. Each line is split into items and the output \(<\text{key, value}>\) pair consists of the item and the value 1. This is the local frequency of the item. The reduce task starts with the itemsets of length 1 and generates candidates with length 2. During step \(k\) of the algorithm it will start with length \(n\) itemsets and generate length \(k + 1\) candidate itemsets. If the reduce task cannot generate bigger candidate itemsets it will stop the whole computation. Frequent itemsets are calculated based on different values of minimum support threshold. Support decision system will check for the appropriate support count value for generating strong association rules.

**3.1. Association Rule Generation**

The output of distributed frequent mining algorithm is frequent itemsets which will be given as input to the association rule generator module to generate strong association rules which satisfies minimum confidence threshold. Association rules can be generated as follows [16].

- For each frequent itemset, \(l\), generate all non-empty subsets of \(l\).
comparatively small execution times, the number of nodes must be increased with increase in the database size. When the size of the database is 1GB respectively. The result shows that the performance of the algorithm depends on the number of nodes and the size of the dataset. For a data set of size 5GB that was distributed on single node, the execution time for the CDA, FDM and MR-DARM algorithms are 5670 seconds, 3680 seconds and 269 seconds respectively. So, in order to obtain comparatively small execution times, the number of nodes must be increased with increase in the database size. It is noticeable that the performance of the algorithms increases with increase in number of nodes, and the proposed algorithm gives much better performance than CDA as well as FDM when the size of the dataset is large.

**Figure 2** Dataset Size Vs Execution Time for Single Node Cluster with 1% Minimum Support
5. CONCLUSION AND FUTURE SCOPE

HDFS and MapReduce play really an important role for handling and analyzing of large datasets. However, most of the algorithms have limitation of processing speed. In this paper, hadoop based distributed approach is presented which process the transactional dataset into partitions and transfers the task to all participating nodes. The purpose is to reduce inter node message passing in the cluster. In preprocessing using Hadoop MapReduce, it has been observed that as the number of reducer increases, the execution time is significantly decreases. The experimental results show that the parallel processing task scales linearly with the number of nodes and the size of the dataset. In this paper, The MR-DARM algorithm is implemented to find distributed frequent itemsets. As the number of node is increased, the performance is really improved by considering lower minimum support factor and large database size. The proposed algorithm generates a smaller candidate set and uses a less message passing than CDA and FDM algorithm, thus the execution time of the proposed algorithm is less as compare to others. The proposed algorithm is more flexible, scalable and efficient distributed frequent pattern mining algorithm for mining large data.
The time efficiency of the algorithm may be improved by using FP-tree based data structures for the candidate itemset generation.

6. ACKNOWLEDGEMENTS

The authors take this opportunity to thank all the researchers from the domain of big data analysis for their immense knowledge and kind support throughout the work. Also would like to thank our institute for their resources and constant inspiration. Special thanks to the authority of AMUL dairy located at Anand district for providing sales dataset. At last heartiest thanks to our family and friends for encouraging us to make this a success.

REFERENCES

