International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 11 (2018) pp. 9403-9406
© Research India Publications. http://www.ripublication.com

EEAAFIT: Enhanced Efficiency of Apriori Approach for Frequent Itemset Techniques

Dr. Ramalingam Sugumara[2] and M. AppasAli[2]

[1]Department of Computer Science, Christhu Raj College, Trichy, India.

[2]Research Scholar, Department of Compute Science, Christhu Raj College, Trichy, India.

Abstract

Data Mining concerns with design and growth of computational algorithms and techniques for discovering hidden patterns and rules which are nontrivial, interesting, previously unknown and potentially useful from data in databases. Data mining association rules is a common method used to find meaningful relationship between large amounts of item sets. Frequent itemset mining is widely used as a fundamental data mining technique. However, as the data size increases, the relatively slow performances of the existing methods. The problem of association rule mining is to discover a set of attributes or items shared among a large number of transactions in a given database. In association rule discovery, it is found how the presence of a set of items in a transaction influences the presence of another set of items in the same transaction and how regularly it happens in the whole database. Apriori is the key algorithm in association rule mining. Many approaches are proposed in past to improve Apriori but the core idea of the algorithm is same but support and confidence of itemsets and earlier studies finds that Apriori is inefficient due to many scans on database. In this paper, we are proposing a method to improve Apriori algorithm efficiency by reducing the database size as well as reducing the time unused on scanning the transactions.

Keywords: Data Mining , Association , Apriori, Itemsets, Algorithms.

INTRODUCTION

Association rules are if and then statements that used to reveal relationships between uncorrelated data in a database, relational database or other information repository [1]. It is used to extract the relationships between the objects data which are frequently used together. Applications of association rules are basket data analysis, storage planning etc. For example, if the customer buys milk then he may also buy bread. There are two significant measures that association rules uses, support and confidence. It describes the relationships and rules created by studying data for frequently used if and then patterns. Association rules are generally required to satisfy a user-defined minimum support and a user–defined minimum confidence.

Support: Support defines the transactions that contains itemset. If p, q are two itemsets, then the support can be defined as the transaction T which defines p / q.

Confidence: Confidence is defined as the percentage of transactions where the itemsets are most likely to occur. If p, q are two itemsets, then, the probability p U q is a subset of transaction, T is called as the confidence.

Frequent patterns algorithms are Apriori algorithm Frequent pattern growth algorithm and Eclat algorithm. These algorithms are used to generates rules on associated attributes.

Apriori algorithm is a two stage process. First, the candidate item set generation and second, the rule generation. Before starting the working procedure of apriori algorithm, the minimum support P is defined by user. Apriori algorithm starts by scanning the complete database, D and find all the frequent items from the database D. First scan the complete database only for 1-itemsets, and then successive iterations deals the 2-itemset. Thus new list of frequent items are created. The process continues untill all the frequent itemsets are extracted from D. Only those frequent items whose minimum support is greater than or equal to P is taken for rule generation [2].

RELATED WORK

This work is based on automobiles study and will help the sellers and customers in making decisions. The objective is to find the important selling factors that affect the relevant sale of vehicles by using the association rule mining algorithm. Most famous algorithm of association rule mining is Apriori is used for knowledge discovery. Research work will improve the existing Apriori algorithm and will reduce some of the drawbacks of the existing algorithm. [3]

An improved algorithm in this paper with a aim of minimizing the temporal and spatial complexities by cutting off the database scans to one by generating compressed data structure bit matrix(b_matrix) and by reducing redundant computations for extracting regular itemsets using top down method. theoritical analysis and experimental results shows that improved algorithm is better than classical apriori algorithm. [4]

This algorithm encountered dense data due to the large number of long patterns emerge, this algorithm's performance declined dramatically. In order to find more valuable rules, this paper proposes an improved algorithm of association rules, the classical Apriori algorithm. Finally, the improved algorithm is verified, the results show that the improved algorithm is reasonable and effective, can extract more value information. [5]

This paper indicates the limitation of the original Apriori algorithm of wasting time for scanning the whole database searching on the frequent itemsets, and presents an improvement on Apriori by reducing that wasted time depending on scanning only some transactions. The paper shows by experimental results with several groups of transactions, and with several values of minimum support that applied on the original Apriori and our implemented improved Apriori that our improved Apriori reduces the time consumed by 67.38% in comparison with the original Apriori, and makes the Apriori algorithm more efficient and less time consuming. [6]

In this paper, implement three variations of Apriori algorithm using data structures hash tree, trie and hash table trie. Trie with hash technique on MapReduce paradigm. To emphasize and investigate the significance of these three data structures for Apriori algorithm on Hadoop cluster, which has not been given attention yet. Experiments are carried out on both real life and synthetic datasets which shows that hash table trie data structures performs far better than trie and hash tree in terms of execution time. Moreover the performance in case of hash tree becomes worst. [7]

This work proposes FDM, a new algorithm based on FP-tree and DIFFset data structures for efficiently discovering frequent patterns in data. FDM can adapt its characteristics to efficiently mine long and short patterns from both dense and sparse datasets. Several optimization techniques are also outlined to increase the efficiency of FDM. An evaluation of FDM against three frequent itemset data mining algorithms, dEclat, FP-growth, and FDM* (FDM without optimization), was performed using datasets having both long and short frequent patterns. The experimental results show signi_cant improvement in performance compared to the FP-growth, dEclat, and FDM* algorithms. [8]

In this paper Improved Apriori algorithm which will help in reducing multiple scans over the database by cutting down unwanted transaction records as well as redundant generation of sub-items while pruning the candidate item sets. The performance of this algorithm is analyzed against the FP Growth algorithm in which there is no generation of candidate set. [9]

The algorithm decreases pruning operations of candidate 2-itemsets, thereby saving time and increasing efficiency. For the bottleneck: poor efficiency of counting support, proposed algorithm optimizes subset operation, through the transaction tag to speed up support calculations. Algorithm Apriori is one of the oldest and most versatile algorithms of Frequent Pattern Mining (FPM). Its advantages and its moderate traverse of the search space pay off when mining very large databases. The algorithm improves Apriori algorithm by the way of a decrease of pruning operations, which generates the candidate 2-itemsets by the apriori-gen operation. Besides, it adopts the tag-counting method to calculate support quickly. So the bottleneck is overcome. [10]

This paper presents a load balancing technique designed specifically for parallel publications applications running on multicore applications. This architecture provides a hardware parallelism through cores inside the CPU. It increased performance low cost as compare to single core machines attracts HPC high performance computing connectivity. [11]

A distributed association rule mining algorithm on Spark named as Adaptive-Miner which uses adaptive approach for finding frequent patterns with higher accuracy and efficiency. Adaptive-Miner uses an adaptive strategy based on the partial processing of datasets. Adaptive-Miner makes execution plans before every iteration and goes with the best suitable plan to minimize time and space complexity. Adpative-Miner is a dynamic association rule mining algorithm which change its approach based on the nature of dataset. Therefore, it is different and better than state-of-the-art static association rule mining algorithms and conduct in-depth experiments to gain insight into the effectiveness, efficiency, and scalability of the Adaptive-Miner algorithm on Spark. [12]

PROPOSED METHOD

In this proposed approach to improve apriori algorithm efficiency, we focus on reducing the scan time consumed for candidate generation. In the process to find frequent item sets, first find the size of a transaction (Size) in Database and also find the maximum data sets. It compared with minimum support and take relevant data item data item set. The remaining data items sets are removed from the database. So size will be reduces and also the scanning time also reduced.

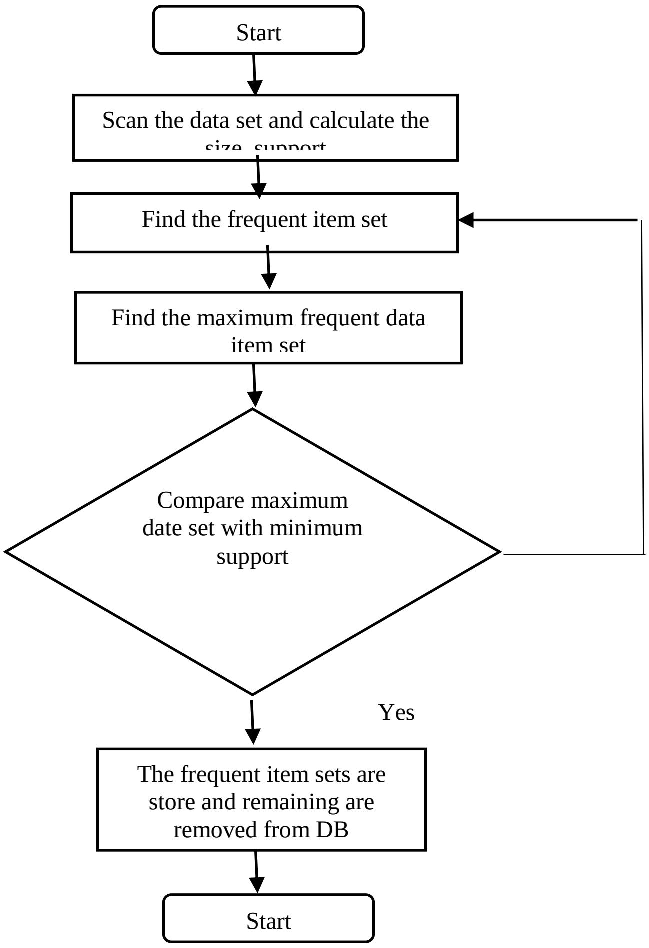
Start
Scan the data set and calculate the size, support
Find the frequent item set
Find the maximum frequent data item set
Compare maximum date set with minimum support
Yes
The frequent item sets are store and remaining are removed from DB
Start

Figure 1: Proposed Method

The table 1 is the transaction database which have 10 transactions

Table 1. Transaction Database

Transaction	Items
T1	I1,I3,I7
T2	I2,I3,I7
T3	I1,I2,I3
T4	I2,I3
T5	I2,I3,I4,I5
T6	I2,I3
T7	I1,I2,I3,I4,I6
T8	I2,I3,I4,I6
T9	I1
T10	I1,I3

By using the above table 1 we calculate the size of the transaction. It is available on table 2.

Table 2. Transaction Database with size

Transaction	Items	Size
T1	I1,I3,I7	3
T2	I2,I3,I7	3
T3	I1,I2,I3	3
T4	I2,I3	2
T5	I2,I3,I4,I5	4
T6	I2,I3	2
T7	I1,I2,I3,I4,I6	5
T8	I2,I3,I4,I6	5
T9	I1	1
T10	I1,I3	2

The table 3 gives the information about the number of items scanned to get 1 frequent itemsets.

Table 3. Transaction Database with support

Item	Transaction	Support
I1	T1,T3,T7,T9,T10	5
I2	T2,T3,T4,T5,T6,T6,T8	7
I3	T1,T2,T3,T4,T5,T6,T7,T8,T10	9
I4	T5,T7,T8	3
I5	T5	1
I6	T7,T8	2
I7	T1,T2	2

The table 3 contains items, itemsets whose support < min_sup are eliminated or removed from the database.

Table 4. Final Transaction Database

Transaction	Items	Size
T5	I2,I3,I4,I5	4
T7	I1,I2,I3,I4,I6	5
T8	I2,I3,I4,I6	5

The table 4 contains items, respective support count and transactions from database.

Table 5. Frequent 3 items

Transaction	Items	Size
T5	I2,I3,I4	3
T7	I2,I3,I4	3
T8	I2,I3,I4	3

Based on the above process to find frequent itemset for a given transaction database. The table 5 contains frequent itemset, association rules are generated from non-empty subsets which satisfy minimum support value.

The number of transactions that are scanned to find the frequent item sets for our given example and below table 6 shows the difference in count of transactions scanned by using the apriori algorithm and our proposed method.

Table 6. Algorithm Comparison

Items	Apriori Method	Proposed Method
1	70	70
2	60	14
3	30	9

The number of transactions scanned is same for both apriori and our proposed method.

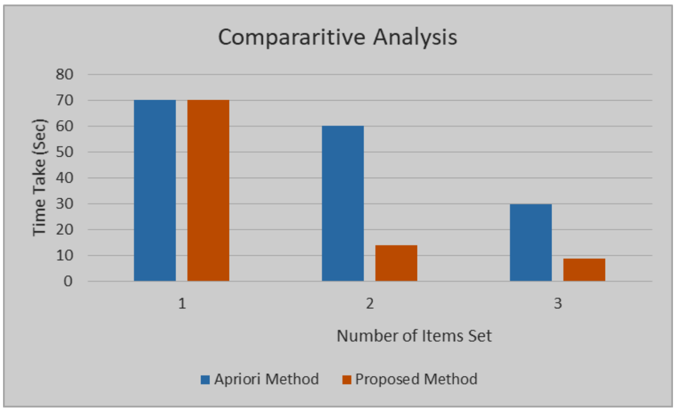

Figure 2: Comparative Analysis

CONCLUSION

The ultimate purpose of this method is to reduce the time taken to scan the database transactions. We find that with increase in value of number of data items, number of transactions scanned decreases and thus, time consumed also decreases in comparison to apriori algorithm. Because of this, time taken to generate candidate sets in our proposed method also decreases in comparison to apriori.

REFERENCES

[1] Yaqiong Jiang, Jun Wang, “An Improved Association Rules Algorithm based on Frequent Item Sets”, 1877-7058 © 2011 Published by Elsevier Ltd, doi:10.1016/j.proeng.2011.08.625,2011

[2] Sheetal Bagde,Anju Singh, “An Efficient Approach for Association Rule Mining by using Two Level Compression Technique”, International Journal of Computer Applications (0975 – 8887) Volume 112 – No 15, February 2015

[3] Dr. Gurpreet Singh, Er. Sonia Jassi , “Implementation and evaluation of optimal algorithms for computing association rule learning”, International Journal Of Engineering And Computer Science ISSN:2319-7242 Volume 6 Issue 7, Page No. 22128-22133 July 2017.

[4] Shalini Dutt, Naveen Choudhary & Dharm Singh, “An Improved Apriori Algorithm based on Matrix Data Structure”, Global Journal of Computer Science and Technology: C Software & Data Engineering Volume 14 Issue 5 Version 1.0 Year 2014.

[5] Jiao Yabing , “Research of an Improved Apriori Algorithm in Data Mining Association Rules”, International Journal of Computer and Communication Engineering, Vol. 2, No. 1, January 2013.

[6] Mohammed Al-Maolegi1, Bassam Arkok , “An Improved Apriori Algorithm For Association Rules” International Journal on Natural Language Computing (IJNLC) Vol. 3, No.1, February 2014.

[7] Sudhakar Singh ,Rakhi Garg, P.K. Mishra,“Performance Analysis of Apriori Algorithm with Different Data Structures on Hadoop Cluster”, International Journal of Computer Applications (0975 – 8887) Volume 128 – No.9, October 2015.

[8] George GATUHA, Tao JIANG, “Smart frequent itemsets mining algorithm based on FP-tree and DIFFset data structures”, Turkish Journal of Electrical Engineering & Computer Sciences, 2017

[9] Sangita Chaudhari, Mayur Borkhatariya, Apurva Churi, Mohini Bhonsle, “ Implementation and Analysis of Improved Apriori Algorithm”, International Journal of Engineering Science and Innovative Technology (IJESIT) Volume 5, Issue 2, March 2016

[10] Darshan M. Tank,” Improved Apriori Algorithm for Mining Association Rules”, I.J. Information Technology and Computer Science, 2014.

[11] Prantik Pancholi, Shital Khairnar, Jyoti kamble , Amol Jadhao, “ MACH: Performance Enhancement in Multi-core Processor using Apriori Algorithm with file Chunking”, International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 - 0056, Volume: 03 Issue: 04 | April-2016.

[12] Sanjay Rathee, Arti Kashyap, “ Adaptive‑Miner: an efficient distributed association rule mining algorithm on Spark”, J Big Data (2018) 5:6 https://doi.org/10.1186/s40537-018-0112-0, 2018.