IMPROVE THE PERFORMANCE OF FREQUENT ITEMSETS USING APRIORI AND FP TREE ALGORITHM

[1]Shrish Mohan Dubey, [2]Deshdeepak Shrivastava, [3]Aditya Dubey
[1]Asstt.Prof, [2] Asstt.Prof [3] Asstt.Prof
[1]Computer Science &Engg,
[1]ITM GOI, Gwalior, India

Abstract: **Today's era is based on IT technologies, so data storage is increasing day by day. Result of that big amount of data stored in databases and warehouses. Therefore the Data mining becomes popular to explore and analyze the databases for finding the the interesting and unknown patterns and rules known as association rule mining. Association rule mining is one of the essential tasks of descriptive technique which can be found meaningful patterns from big collection of data. Mining frequent item set is basic principle of association rule mining. Many algorithms have been proposed from last many years including Efficient Mining of Frequent Item Sets on Large Uncertain Databases. An efficient Approach for the implementation of FP Tree computes the minimum-support for mining frequent patterns. Now a day, various techniques face the problem of data redundancy, candidate generation, memory consumption problem (FP-tree Algorithms) and other frequent patterns problem. Because of retailer industry many transactional databases include same set of transactions many times, to apply this thought, in this paper present a new technique which is combination of maximal Apriori (improved Apriori) and FP-tree techniques that assure better performance as compare to classical Aprioi algorithm. Another approach is to analyze the many existing techniques for mining frequent item sets and estimate the performance of new techniques and compare with the existing classical Apriori and FP- tree algorithm.**

Keywords:- *Apriori Algorithm,FP, Algorithm,Data Mining,FP Tree*

I. INTRODUCTION

Association rule mining finding frequent patterns, associations, correlations, or structures among sets of items or objects in transaction databases, relational databases. The techniques for discovering association rules from the data have traditionally focused on identifying relationships between items telling some aspect of human behavior, usually buying behavior for determining items that customers buy together.

Let X, Y be a set of items, an association rule has the form :- X→Y.
X is called the antecedent and Y is called the consequent of the rule where, X,Y is a set of items is called as an itemset or a pattern.

The support of an association rule is the support of the union of X and Y :-

Support(X→Y)=(X ∪ Y)/D , where D is containing itemsets X and Y .

The Confidence of an association rule is defined as:-

Confidence (X→Y) = P(X/Y)=Support (X Y)/support (X) Association rule mining is to identify all rules meeting user specified constraints such as minimum support and minimum confidence.

The key step of association mining is frequent itemset (pattern) mining, which is to mine all itemsets satisfying user specified minimum support.
To improve the performance of the rule discovery algorithm, mining association rules may be decomposed into two phases:
1. Discover the large itemsets.
2. Use the large itemsets to generate the association rules for the database that have confidence above a predetermined minimum threshold.

Frequent Itemsets
In order to searching the frequent itemsets, the support of each itemsets must be computed by scanning each transaction in the dataset. A brute force approaches used for this purpose.

The two typical strategies is adopted by these algorithms:-

1.*Apriori *techniques
Is an effective pruning strategy to reduce the combinatorial search space of candidate itemsets .

2.*FP-tree techniques*
The second strategy is to use a compressed data representation to facilitate in-core processing of the itemsets.

Frequent Itemsets for Association Rules

The frequent itemset mining is motivated by problems such as market basket analysis.

An association rule mined from market basket database states that if some items are purchased in transaction and some other items are purchased as well.

The problem of mining frequent itemsets are essentially, to discover all rules, from the given transactional database that have support greater than or equal to the user specified minimum support.

FP-Growth Algorithm

FP-Growth: allows frequent itemset discovery without candidate itemset generation. Two step approach:

Step 1: Build a compact data structure called the FP-tree Built using 2 passes over the data-set.

Step 2: Extracts frequent itemsets directly from the FP-tree

II. Problem Statement

A transaction is called to be maximal frequent if its length is greater than or equal to all other existing transactional patterns(length) and also count the number of occurrences (support) in database is greater than or equal to specified minimum support threshold value .

An itemset is said to be frequent if it support is greater than or equal to given minimum support threshold value i.e. Count > Min(Support).

In our perception the transactional database and minimum support threshold is to be assume , therefore the problem is to find the complete set of frequent itemsets from type of transactional databases is to increase the business, so how can be related customer behavior to find between various items.

III. Research Objective

The main objective of the research is to develop and propose a scheme for mining the association rules out of transactional data set.

The proposed scheme is based on two approaches: Improved Apriori approach and FP-Growth approach.

The proposed scheme is more efficient than Apriori algorithm and FP-growth algorithm, as it is based on two of the most efficient approaches. To achieve the research objective successfully, a series of sequence progresses and analysis steps have been follows:

1. Market Based Analysis

2. Objective of Market Analysis

3. Data Assembling

4. Implementation of Mining Algorithm (Apriori and FP algorithm)

IV. Research Approach

We can summarize the main contribution of this research as follows:
1. To study and analyze these two approaches to mine frequent item sets.
2. To devised a new better scheme Apriori and FP-tree alone using maximal Apriori and FP-tree as combined approach for mining frequent item sets.

V. Review of Literature

A High Performance Frequent Itemset Mining Algorithm Using Confidence Frequent Pattern Tree Kun-Ming Yu, Bin-Chang Wu 2008 3rd International Conference on Innovative Computing Information and Control.

The author proposed a new tree structure to store all identified widely wide-spread item sets and a header table to create a usual item linking list in order to avoid repeating the calculation of known frequent items to speed up the data mining process. However, not many studies concentrate on using known frequent item sets to increase system performance, various processing methods for association data mining are presently being looked into. Most of them focus on data structure and computation improvement. The data structures usually have a high degree of data compression ratio and can express the original information from the database with integrity.

Efficient Mining of Frequent Item Sets on Large Uncertain Databases

Liang Wang, David Wai-Lok Cheung, Reynold Cheng, Sau Dan Lee, Xuan S. Yang IEEE Transactions on Knowledge and Data Engineering 2012.

The author proposed an approximate algorithm, which can efficiently and accurately discover frequent item sets in a large uncertain database and also study the important issue of maintaining the mining result for a database that is evolving. Specifically, the propose incremental mining algorithms, which enable Probabilistic Frequent Item set (PFI) results to be refreshed. This reduces the need of re-executing the whole mining algorithm on the new database, which is often more expensive and unnecessary. We examine how an existing algorithm that extracts exact item sets, as well as our approximate algorithm, can support incremental mining. All our approaches support both tuple and attribute uncertainty, which are two common uncertain database models. And also perform extensive evaluation on real and synthetic data sets to validate our approaches.

Computing the minimum-support for mining frequent patterns

Shichao Zhang, Xindong Wu, Chengqi Zhang, Jingli Lu Knowl. Inf. Syst.2008

In this paper author propose a computational strategy for identifying frequent item-4 sets, consisting of polynomial approximation and fuzzy estimation. More specifically, our 5 algorithms (polynomial approximation and fuzzy estimation) automatically generate actual 6 minimum-supports (appropriate to a database to be mined) according to users' mining re-7 requirements. and experimentally examine the algorithms using different datasets, and demon-8 state that our fuzzy estimation algorithm fittingly approximates actual minimum-supports 9 from the commonly-used requirements.

Mining Frequent Itemsets from Uncertain Data

Chun Kit Chui, Ben Kao, Edward Hung PAKDD2007

This paper proposed study of problem of mining frequent itemsets from uncertain data under a probabilistic framework. And consider transactions whose items are associated with existential probabilities and give a formal definition of frequent patterns under such an uncertain data model. And Also show that traditional algorithms for mining frequent itemsets are either inapplicable or computationally inefficient under such a model data trimming framework is proposed to improve mining efficiency. Through extensive experiments, we show that the data trimming technique can achieve significant savings in both CPU cost and I/O cost.

FP-growth Tree for large and Dynamic Data Set and Improve Efficiency

Rahul Moriwal 2014

In this paper author proposed the improved FP-growth method is an efficient algorithm to mine frequent patterns, in spite of long or short frequent patterns. By using compact tree structure and partitioning-based, divide-and-conquer searching method, it reduces the search costs substantially. But just as the analysis in Algorithm, in the process of FP-tree construction, it is a strict serial computing process. Algorithm performance is related to the database size, the sum of frequent patterns in the database.

An Efficient Approach for the Implementation of Fp Tree:-

SADHANA KODALI 2013

In this paper author proposed an approach for improving the performance of the FP tree using the parameter average support count. Frequent pattern mining is one of the most common mining techniques to identify the frequent patterns in large data sets. The Apriori algorithm is one algorithm very efficient for mining frequent patterns. But the drawback is it generates a number of candidate item sets. The FP tree is a frequent pattern technique without candidate item set generation.

VI. METHODOLOGY USED

1. To analyze the various existing techniques and find their strengths and weakness.
2. To compare the existing techniques.
3. Build a program for our desired problem by using Apriori technique and FP-tree structure.
4. Validate the program by desired input.

VII. Possible Outcome

Dataset which contains the maximal frequent itemset in large amount shows better result with Apriori and FP algorithm.
In the dataset there are many transaction consider which occurs repeatedly in the database and some transaction occur greater than the minimum support.

VIII. Conclusion

1. In this thesis, we considered the following factors for creating our new scheme, which are the time and the memory consumption, these factors are affected by the approach for finding the frequent itemsets.
2. Work has been done to develop an algorithm which is an improvement over Apriori and FP-tree with using an approach of improved Apriori and FP-Tree algorithm for a transactional database.
3. According to our observations, the performances of the algorithms are strongly depends on the support levels and the features of the data sets (the nature and the size of the data sets).

IX. Research Gap

In traditional algorithms for mining frequent itemsets are either inapplicable or computationally inefficient under such a model data trimming framework is proposed to improve mining efficiency. The drawback is it generates a number of candidate item sets. The FP tree is a frequent pattern technique without candidate item set generation . In this algorithm Our focus on developing data structure and computation improvement of them.

X. Reference :-

[1] C.Borgelt. “Efficient Implementations of Apriori and Eclat”. In Proc. 1st IEEE ICDM Workshop on Frequent Item Set Mining Implementations, CEUR Workshop Proceedings 90, Aachen, Germany 2003
[2] C. Borgelt. “An Implementation of the FP- growth Algorithm”. Proc. Workshop Open Software for Data Mining, 1–5.ACMPress, New York, NY, USA 2005..An Implementation of the FP-growth Algorithm.
[3] S.P Latha, DR. N.Ramaraj. “Algorithm for Efficient Data Mining”. In Proc. Int’l Conf. on IEEE International Computational Intelligence and Multimedia Applications, 2007, pp. 66-70.
[4] Chui, Chun-Kit, Ben Kao, and Edward Hung. "Mining frequent Itemsets from uncertain data." Advances in knowledge discovery and data mining (2007): 47-58..
[5] Guo, Yunkai, Junrui Yang, and Yulei Huang. "Fast Updating Maximal Frequent Itemsets Based on Full Merged Sorted FP-Tree." Wireless Communications, Networking and Mobile Computing, 2008. WiCOM'08. 4th International Conference on. IEEE, 2008
[6] Zhang, Shichao, et al. "Computing the minimum-support for mining frequent patterns." Knowledge and Information Systems 15.2(2008): 233-257.
[7] Wang, Liang, et al. "Efficient mining of frequent item sets on large uncertain databases." IEEE Transactions on Knowledge and Data Engineering 24.12 (2012): 2170-2183.
[8] Moriwal, Rahul. "FP-growth tree for large and dynamic data set and improve efficiency." J. Inform. Compute. Science 9 (2014): 83-90.
[9] Shah, Arpan H., and Pratik A. Patel. "Optimum Frequent Pattern Approach for Efficient Incremental Mining on Large Databases using Map Reduce." International Journal of Computer Applications 120.4 (2015).
[10] Shah, Arpan H. , and Pratik A. Patel. "Optimum Frequent Pattern Approach for Efficient Incremental Mining on Large Databases using Map Reduce." International Journal of Computer Applications 120.4 (2015).