International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 3 Issue 1, January - 2014

Dynamic Fp-growth Tree Mining Approach with Projection Technique

Keshav lodhi
M-tech sati vidisha

Dr. R. C. Jain
Director –sati vidisha

Prof. Deepak Sain
Faculty of IT dept.

Abstract

In Data mining is about analyzing data; for information about extracting information out of data. It is a very actual and interesting issue having more and more data stored in database. The most important usage: customer behavior in market purchasing, shopping cart processed information provide, management of campaign , customer relationship management, mining about web usage called web mining, mining of text. In the current age of science we developed such technology by using it each type of data related to anything such like person, place, shop, or any organization can be stored. This data is a big source of information. Data mining provides such techniques that convert stored data into the useful information data mining is the method for extracting valid, accurate and actionable information from database. The Frequent Pattern tree method is best algorithm in association mining by which we can construct frequent patterns in data. By using compact tree structure and applying sectioning-based divide-and-conquer data mining searching method, we can reduce or minimize the costs search probably. It is just as the examination of much CPU system or by reducing computer memory for solving problem. By this technique can be apparently decrease or optimize the costs for interchanging and monopolizing control information and the complexity of algorithm is also decreased in efficient manner. Here we will discuss on balanced dynamic fp tree construction technique or method that will optimize or reduce the cost of construction and execution and also show better improvement in performance as compare to other methods.

1. Introduction

We in the last decade the sizes of databases has increased rapidly and in the future it will increased more rapidly. This has led to interest in the development of tools capable in the automatic extraction of useful information from data. The term Data Mining, or Knowledge Discovery in large Databases, has been adopted for a field of research dealing with the automatic discovery of implicit information or knowledge within databases a very influential association rule mining algorithm, Apriori has been developed for association rule mining in large transaction databases. A big step forward in developing the working or performance of these algorithms was made by the frequent pattern tree. Frequent-pattern mining contributes very vital role in mining association correlations, causality, sequential patterns, max-patterns, episodes mining, multi-dimensional patterns, contrast patterns, and several other substantial data mining work. In the many earlier work, as in [1] , [2] and [3] used an approach, which is similar to Apriori and also rely on the similar property of Apriori heuristic called anti-monotone property.

The Apriori heuristic is achieved exceptional performance output by minimizing the proportions of candidate sets. Different cases have different scenario, some situations have a large number of frequent patterns, long patterns, or with very low minimum support thresholds, those algorithms which have similar technique like Apriori method may undergo from the two different types of cost:

By doing all kind of management to handle a huge number of candidate sets but it has still more cost. In Apriori like algorithms generates large number of candidates so in practically if the database is too big and it can not fit into the main memory. These are two points where we will concentrate. In many situations, it is really hard to do the rescanning of dataset and test a huge set of candidates based on pattern match. Can we develop a technique that does not produce candidate in repeatative way? And apply some different data mining technique (in terms of data structure) to keep the cost low in FP mining? In our paper, we develop some useful techniques to resolve the problem. At First, a solid data structure as frequent-pattern tree is generated, for the confirmation that the tree structure we used is informative and compact, only items of frequent length – one will have nodes in the tree, the FP-Tree nodes are managed in such way that more repeatedly occurring nodes will have greater opportunity to share a node rather than lower frequently

occurring ones. In the next technique pattern-fragment growth mining procedure is utilized, that is based on fp tree and begin from a frequent le pattern of length-one (initial suffix pattern), checks its conditional-pattern base (a "sub-database" that has the set of frequent items co-occurring with the suffix pattern), generate its own Frequent Pattern tree, perform mining work recursively with similar tree. Third, we used a search method in mining is different than Apriori-or other similar candidate generation. It is a divide-and conquer technique (partitioning based) that suddenly decrease the magnitude of sub database constructed at the succeeding step of research as well as the magnitude of its relevant conditional Frequent Pattern tree provides .

2. Base Notation and Implementation

A condition that may occur in a very large database. To project the database, a projection of database technique is introduced to deal in the condition when the Frequent Pattern tree cannot be put in main memory Extensive experimental results have been reported. Experimental result will shows some point at where the size of Frequent Pattern Tree and important point of frequent pattern growth on the projection of data to generate Frequent Pattern Tree. The shared parts can be merged by applying single prefix structure until count registration is done. On the basis of order of frequent items two or more record accord a common prefix.

FP-tree

FP-Tree, frequent pattern mining, in data mining mining breaks the Apriori bottlenecks problem. The frequent item-sets are generated with only two passes over the database and without any candidate generation process. FP-Tree shows much better results than Apriori the reason is that the support threshold (minimum) goes low, length of frequent item-sets and the number of frequent items enlarge dramatically. Development of a solid Frequent Pattern tree guarantees that succeeding mining can be done with a solid data structure. This does not give ensure by itself that it will be most efficient since one may still find out the problem making combination of candidate generation .

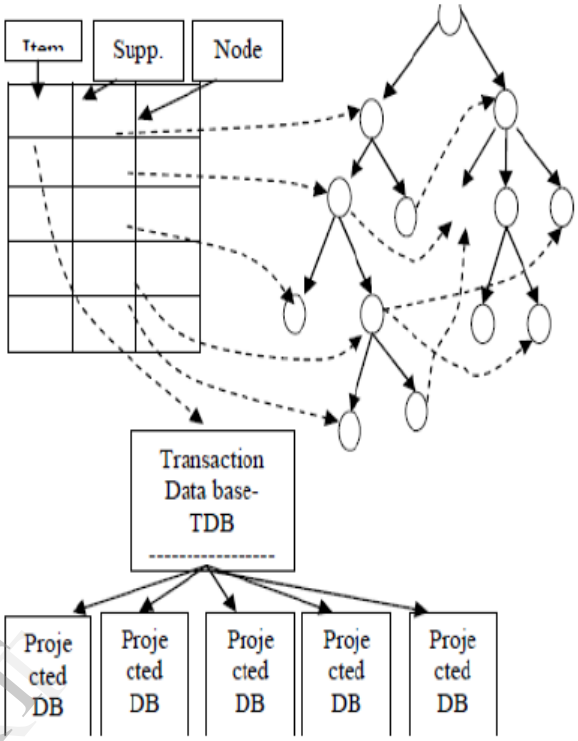

Figure 1 FP tree data structure of database

1. COMPARIOSION ON THE BASIS OF EXECUTION TIME ANDMINIMUM SUPPORT COUNT BETWEEN FPGROWTH

Tree conditional pattern, FP-GROWTH TREE with DB parallel projection and FP-GROWTH Tree with Database Partition projection.

TABLE 1 COMPARISION OF TECHNIQUES

Number of records	Time taken to execute (in mili second) FP-Growth tree with conditional	Time taken to execute (in mili second) FP-Growth tree with database parallel projection	C Time taken to execute (in mili second) FP-Growth tree with database partition
2	100	132	151
3	83	121	132
4	58	83	94
5	44	72	78
6	39	64	68

Graph representing the comparison of FP-GROWTH Tree, Data base Parallel Projection and Data base Partition projection when number of records varying.

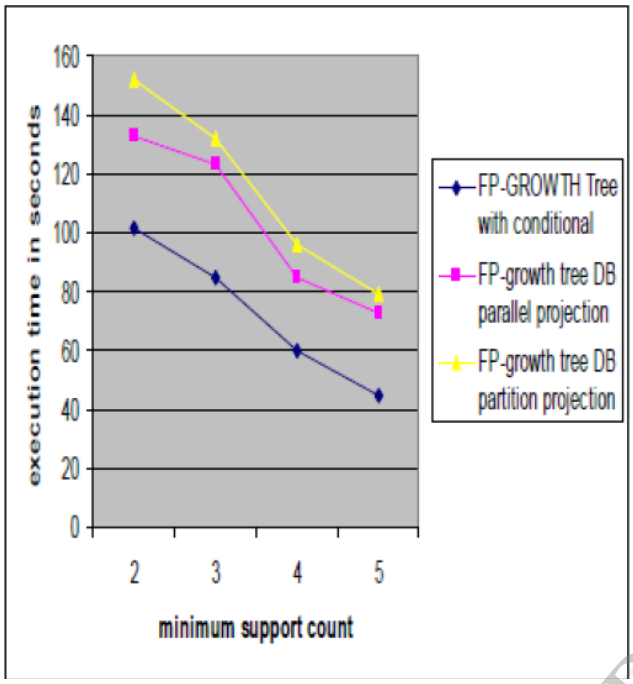

Figure 2 Comparison in execution time

COMPARIOSION ON THE BASIS NUMBER OF RECORDS AND MINIMUM SUPPORT COUNT BETWEEN FP-GROWTH TREE Conditional pattern, FP-GROWTH Tree with DB parallel projection and FP-GROWTH Tree with Database Partition projection.

TABLE 2 Comparison of techniques in respect of records

Number of records	Time taken to execute (In millisecond) FP-GROWTH Tree with conditional	Time taken to execute (In millisecond) FP-GROWTH Tree with Data base Parallel Projection	Time taken to execute (In millisecond) FP-GROWTH Tree with Data base Partition projection
200	64	84	91
300	68	96	101
400	129	143	154
500	186	193	220

GRAPH REPRESENTING THE COMPARISON OF FP-GROWTH TREE, DATA BASE PARALLEL PROJECTION AND DATA BASE PARTITION PROJECTION WHEN MINIMUM SUPPRT COUNT VARING.

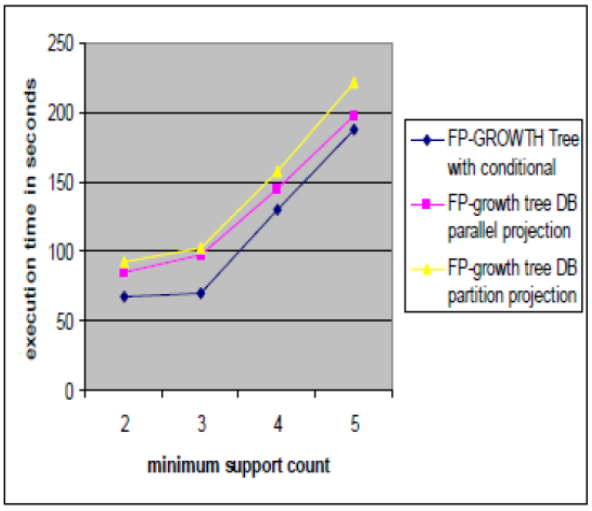

Figure 3 Comparison in approach

This practical work leads to the following conclusions.

(1.) The advantage of FP-tree is that it is more space efficient in case of dense datasets .A dense dataset can be compressed many times.

(2.) The FP-tree gets bushy in such cases, Fp-tree size may be increased due overhead of links. Degree of sharing in branches of frequent pattern tree becomes low. That's why we create a projected database in place of Frequent Pattern Tree. We only create an fp tree when the datasets cross the certain density limit.

(3.) From the practical work, a point came out that such a threshold is pretty low. So, we can introduce fp-tree even for very large and/or sparse database, after one or a few rounds of database projection.

(4.) FP-Growth Tree is more efficient then tree projection but it is difficult to maintain it in memory so tree projection is used in tree projection two types of projection is used parallel projection is good but it takes more memory but partition projection takes more time is execution but takes less apace compare to parallel projection .

3. Advantages

(1) To reduce the cost of multiple scan in succeeding mining processes a parallel frequent pattern tree is generated. This tree will be smaller in size then the actual database. Because its size is small then it will take less time to scan.

(2.) By adjoining projection technique into the process of tree construction, we save the costly frequent items main

scans in which hugely shorten the time of tree-construction. And the performance is much better than the FP-tree method.

(3) The costly candidate propagation and test is avoided by applying a pattern growth based technique.

(4) By applying partioning-based techniques, which greatly help to keep the size of the succeeding conditional PFP trees pattern bases and conditional pattern bases low. I have proposed a decent method for constructing projection frequent pattern tree, and if we conjoined the method of constructing PFP-tree and the PFP-tree-based mining, we can mine frequent patterns efficiently in large databases[9][10]. We call this conjoined method PFP growth [11].

4.Conclusion

Since a transactional databases are projected and the list of frequent items is mined in reverse order. First we mined the least frequent item from the parallel projected database and continue further. In this paper we have introduced a novel approach and it shows significant improvement over the results. To see the actual performance all the three techniques should be tested under the same environment. Because it may differ in various ways, some time the dataset is differ, sometimes the input/output devices perform differently. We have provided a console based application in which there is no GUI control. For the future enhancement the application can be developed for a 64-bit operating system. In this approach the selection of procedure (parallel or partition) is not dynamic. The user will decide which type of technique he wants to use.

REFERENCES

[1] NAVATHE, A., OMIECINSKI, E., AND SAVASERE, S. AN EFFICIENT ALGORITHM FOR MINING ASSOCIATION RULES IN LARGE DATABASES. PROCEEDINGS OF THE VLDB CONFE. 1995.

[2] AGRAWAL, R. AND SRIKANT, R. FAST ALGORITHMS FOR MINING ASSOCIATION RULES. VERY LARGE DATABASE, 487-499.

[3] YIN, J., PEI, J., AND HAN, Y. MINING FREQUENT PATTERNS WITHOUT CANDIDATE SET GENERATION. SIGMOD, 2000.

[4] BRIN, S., MOTWANI, R., ULLMAN JEFFREY D., AND TSUR SHALOM. GENERALIZING ASSOCIATION RULES TO CORRELATIONS. SIGMOD 26[2], 265-276 1997.

[5] SILVERSTEIN, S., MOTWANI, R., AND BRIN, C. BEYOND MARKET BASKETS: GENERALIZING ASSOCIATION RULES TO CORRELATIONS. SIGMOD 26[2], 265-276. 1

[6] HAN, J., PEI, J., MORTA ZAVI-ASL , B., CHEN, Q., HSU, U., AND DAYAL, M.-C. FREESPAN: FREQUENT PATTERN-PROJECTED SEQUENTIAL PATTERN MINING IN ACM SIGKDD , 2000.

[7] A PRIORI KNOWLEDGE AND HEURISTIC REASONING IN ARCHITECTURAL DESIGN PETER G. ROWE JAE VOL. 36, NO. 1 (AUTUMN, 1982).

[8] ORLANDO, PEREGO S., PALMERINI, P., ENHANCING THE APRIORI ALGORITHM FOR FREQUENT SET COUNTING. IN 3RD INTERNATIONAL CONFERENCE ON DATAWAREHOUSING AND KNOWLEDGE DISCOVERY. 2001.

[9] PIATETSKY-SHAPIRO, G., FAYYAD, U., AND SMITH, P., "FROM DATA MINING TO KNOWLEDGE DISCOVERY: AN OVERVIEW," IN FAYYAD, U., PIATETSKY-SHAPIRO, G., SMITH, P., AND R. (EDS.) ADVANCES IN KNOWLEDGE DISCOVERY AND DATA MINING AAAI/MIT PRESS, 1996, PP. 1-35.

[10] RELUE HUANG, H., WU, X., R. ASSOCIATION ANALYSIS WITH ONE SCAN OF DATABASES. IN THE 2002 IEEE INTERNATIONAL CONFERENCE ON DATA MINING. 2002.

[11] LIU ,TANG ,WANG, K., L., HAN, J., J. TOP DOWN FPGROWTH FOR ASSOCIATION RULE MINING. PROC.PACIFIC-ASIA CONFERENCE, PAKDD 2002, 334-340. 2002.

[12]ZAKI, M. J. ,HSIAO, C.-J. CHARM: AN EFFICIENT ALGORITHM FOR CLOSED ITEMSET MINING. SIAM [INTERNATIONAL CONFERENCE ON DATA MINING.2002