International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435(Online) Volume 2, Issue 6, June 2014)

Intensification of Execution of Frequent Item-Set Algorithms

Ritu[1], Jitender Arora[2]

[1]M. Tech schacolr, RIMT, Chidana
[2] H.O.D. CSE Dept, RIMT, Chidana

***Abstract*— Data mining is an important area for the researcher to do more work in advance. There are huge amount of data in organization which need to organize in a pattern that make it easy to discover. Knowledge discovering in data mining (KDD) is an important process which provide a way to access data in easy way using clustering, classification and association rules. So in our work we are going to implement APRIORI algorithm and FP-Growth algorithm. Main propose of our work is to improve in FP-Growth algorithm and compare with both.**

***Keywords*— Data Mining, Apriori Algorithm, FP-Algorithm, Extended FP-Growth.**

I. INTRODUCTION

Due to the wide availability [1] of huge amount of data and the imminent need for turning such data in to useful information and the knowledge, data mining has attracted a great deal of attention in information industry. The information and knowledge gained can be used for applications ranging from market analysis, fraud detection, and the customer retention. Knowledge Discovery in Databases (KDD) is the automated extraction of novel, understandable and potentially useful patterns implicitly stored in large databases repositories. Data mining is an essential step in the process of knowledge discovery in databases, in order to extract patterns. Thus data mining refers to extracting or mining knowledge [2] from large amounts of data.

Knowledge discovery consists of the following steps:

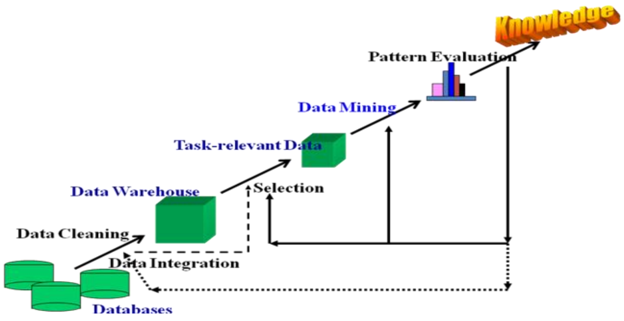

Fig 1 Data mining as a step in Knowledge Discovery

- Data Cleaning: To remove noise and inconsistent data.
- Data integration: Where multiple data sources may be combined.
- Data Selection: Where data relevant to the analysis task are retrieved from the database.
- Data mining: An essential process where intelligent methods are applied in order to extract data patterns.
- Pattern evaluation: To identify the truly interesting patterns representing knowledge based on some interesting measures.
- Knowledge presentation: Where visualization and the knowledge representation techniques are used to present the mined knowledge to the user.

Thus data mining is only the one step in the entire process which is essential because it uncovers hidden patterns for evaluation. Data mining tasks to find these various patterns include:

a) Characterization: Data characterization is a summarization of the general characteristics or features of a user-specified target class of data. For example, the user may like to characterize software products whose sales increased by 10% in the last year. The output of data characterization can be presented in various forms like data cubes.

b) Discrimination: Data discrimination is a comparison of the general features of a user- specified target class data objects with the general features of objects from one or a set of (user-specified) contrasting classes. For example, the user may like to compare the general features of software products whose sales increased by 10% in the last year with those whose sales decreased by at least 30% during the same period.

c) Association Analysis: Association analysis is the discovery of association rules showing attribute-value conditions [3] that occur frequently together in a given set of data. Association analysis is widely used for market basket or transaction data analysis and forms the subject matter of this thesis.

d) *Classification and Regression:* Classification is the processes of finding a set of models that describe and distinguish data classes or concepts, for model to predict the class of objects whose class label is unknown. While classification predicts is applied if the field being predicted comes from a real-valued domain.

e) *Cluster Analysis:* Objects in a database are clustered or grouped based on the principle of maximizing intra class similarity and minimizing interclass similarity. Market segmentation for identifying common traits of groups of people, discovering new types of stars in datasets.

f) *Outlier Analysis:* Outliers are data objects that do not comply with the general behavior or model of the data. The analysis and mining of outliers is crucial.

g) *Evolution Analysis:* Data evolution analysis describes and models regularities or trends for objects whose behavior changes over time. Although this analysis may include any of the above functionalities on time-related data, distinct features of such an analysis include time-series data analysis, sequence or periodicity.

In general, data mining tasks can be classified into two categories: descriptive and predictive. Descriptive mining tasks characterize general properties of the data in the database. Examples include association rule discovery and clustering. On the other hand, predictive mining tasks perform inference on the current data in order to make predictions. Examples of predictive mining tasks include classification and regression. I do my work on associations rule algorithms for finding association rule.

II. Apriori Algorithm In Data Mining

The Apriori algorithm is firstly purposed by R.Agrawal and R.Srikant in 1994 for mining frequent item-set for Boolean association rule [5]. The basic association rule mining performed by apriori is:

- Find the frequent item-sets: the sets of items that have minimum support. A subset of a frequent item-set must also be a frequent item-set i.e., if {AB} is a frequent item-set, both {A} and {B} should be a frequent item-set Iteratively find frequent item-sets with cardinality from 1 to k (k-item-set)
- Generate association rules using frequent item-set.

Apriori employs an iterative approach known as level-wise search, where k-item-set are used to explore (k+1) item-set.

In this firstly the algorithm scans the database to find the set of frequent 1-item-set and do count for each item and collecting those items that satisfy the minimum _support. The resulting set is denoted by L1.Next L1 is used to find L2(frequent 2-item-set) which is used to find L3 and so on, until no more frequent k-item-sets can be found. The finding of each k requires one full scan of a database. Apriori has monotonicity property which states-all non-empty subsets of a frequent item-set must also be frequent.

Key Concepts

- Frequent Item-sets: The sets of item which has minimum support.
- Apriori Property: Any subset of frequent item-set must be frequent.
- Join Operation: To find Lk, a set of candidate k-item-sets is generated by joining Lk-1with itself.

III. FP-Growth Algorithm

As there are two disadvantaging of Apriori-like method:

- It may need to generate a huge number of candidate sets. For example, if there are 10^4 frequent 1-item-sets, the Apriori algorithm will need to generate more than 10^7 candidate 2-item-sets and accumulate and test their occurrence frequencies.
- It may need to repeatedly scan the database and check a large set of candidates by Pattern matching.

Therefore, we came up with a solution that is finding the frequent item-set without candidate generation: FP-GROWTH approach is based on divide and conquers strategy for producing the frequent item-sets. FP-growth is mainly used for mining frequent item-sets without candidate generation. Major steps in FP-growth is-

Step1- It firstly compresses the database showing frequent item-set in to FP-tree. FP-tree is built using 2 passes over the dataset.

Step2: It divides the FP-tree in to a set of conditional database and mines each database separately, thus extract frequent item-sets from FP-tree directly.

It consists of one root labelled as null, a set of item prefix sub trees as the children of the root, and a frequent .item header table. Each node in the item prefix sub tree consists of three fields: item-name, count and node link where--- item-name registers which item the node represents; count registers the number of transactions represented by the portion of path reaching this node, node link links to the next node in the FP- tree.

Each item in the header table consists of two fields---item name and head of node link, which points to the first node in the FP-tree carrying the item name.

IV. Implementation Result

We implement our proposed algorithm in MATLAB tool. GUI of MATLAB is used to provide comparison between APRIORI, FP-Growth, and Extended FP-Growth [4]. I run implementations of apriori, FP-Growth and Enhanced-FP on three datasets mushroom, pumsb and T40I10D100K. Graphs show the execution time of implementations over the various supports. The blue line refers to apriori algorithm. The red line refers to FP-Growth algorithm. The green line refers to Enhanced-FP.

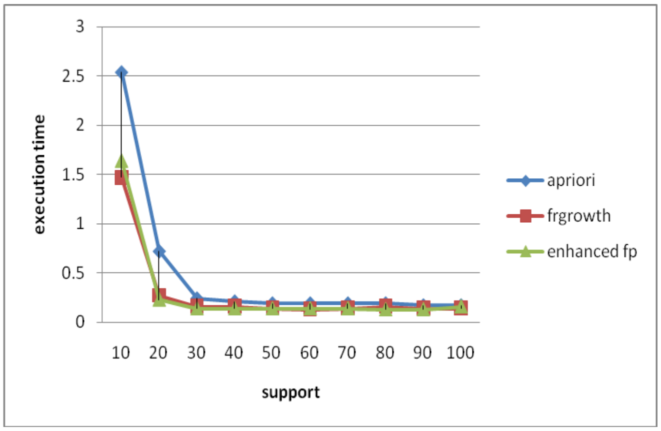

Fig 2 Results of Apriori, FP-Growth and Enhanced-FP on mushroom

Figure 2 Shows the running time of the compared algorithms on mushroom data with different minimum supports represented by percentage of the total transactions. Under minimum supports, Enhanced-FP run faster than FP-Growth and Apriori as well as Enhanced-FP run faster as the support grows. Thus on the dataset mushroom performance of Enhanced-FP is better than Apriori, FP-Growth

Figure 3 Shows the running time of the compared algorithms on pumsb data with different minimum supports represented by percentage of the total transactions. Under minimum supports, Enhanced-FP run faster than FP-Growth and apriori as well as Enhanced-FP run faster as the support grow. While under small minimum supports performance of apriori is better. Thus on the dataset pumsb performance of Enhanced-FP is better than Apriori,FP-Growth.

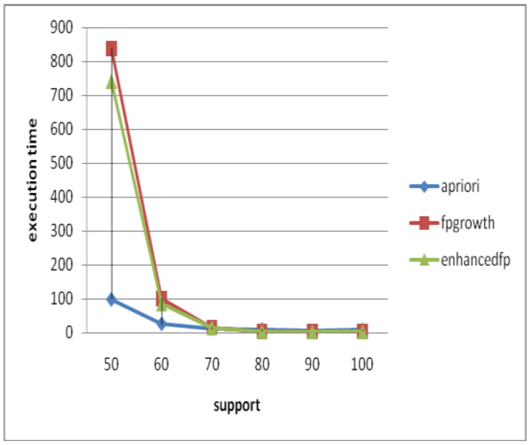

Figure 4 Fig6.2: Results of Apriori, FP-Growth and Enhanced-FP on pumsb.tab

Figure 15 shows the running time of the compared algorithms on T40I10D100K data with different minimum supports represented by percentage of the total transactions. Under minimum supports, Enhanced-FP run faster than FP-Growth and Apriori as well as Enhanced-FP run faster when supports grows. Thus on the dataset T40I10D100K performance of Enhanced-FP is better than Apriori, FP-Growth.

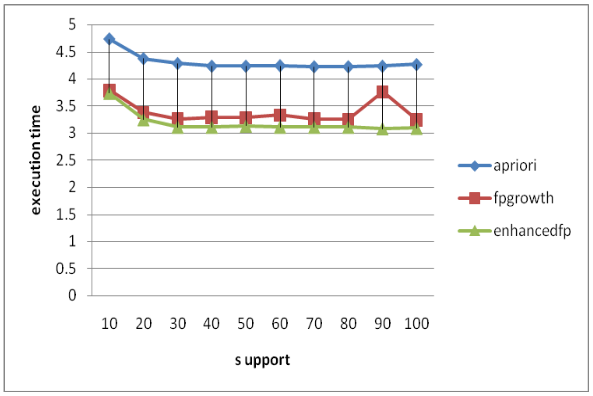

Figure 5 Results of Apriori,FP-Growth and Enhanced-FP on T40I10D100K.dat

V. Conclusion

The FP-Growth algorithm overcomes the two major disadvantage of apriori that is multiple scan of database and candidates generation. It does its work in two passes. Firstly construct the FP-tree and then generate conditional FP-trees from it.

From these it will generate the frequent item-set. But the main problem in this is that, in this construction of FP-tee is very expensive. We cannot count the support until the complete FP-tree is constructed. Thus it works well than apriori or we can say that faster than apriori.

REFERENCES

[1] Han J W, Kamber M. Data Mining: Concepts and Techniques. SanFrancisco: CA. Morgan Kaufmann Publishers.2001:2~4.

[2] Goethals B., “Survey on Frequent Pattern mining,”MANUSCRIPT, 2003.

[3] S. Kotsiantis, D. Kanellopoulos, Association Rules Mining: A Recent Overview, GESTS International Transactions on Computer Science and Engineering, Vol.32 (1), 2006, pp.71-82.

[4] Said A., Dr. Dominic P., Dr. Abdullah A., “A Comparative Study of FP- Growt Variations ” Proc. IJCSNS International Journal of Computer Science and Network Security, VOL.9 No.5 (2009).

[5] Bodon F. “A fast apriori implementation”. In Goethals B. and Zaki M.J., editors, Proceedings of the IEEE ICDM Workshop on Frequent Itemset Mining Implementation (FIMI’03), CEUR Workshop Proceedings, Melbourne, Florida, USA, Vol. 90,(2003).