IAFP: Integration of Apriori and FP-Growth Techniques to Personalize Data in Web Mining

Manasa G[*], Mrs. Kulkarni Varsha[**]

[*]M.Tech, Department of Computer Science & Engineering, Sri Venkateshwara College of Engineering, Bangalore
[**]Assistant Professor, Department of Computer Science & Engineering, Sri Venkateshwara College of Engineering, Bangalore

Abstract- Mining frequent patterns in transaction databases, time-series databases, and many other kinds of databases has been studied popularly in data mining research. Most of the studies adopt an Apriori-like candidate set generation-and-test approach. However, the candidate set generation is still costly, especially when there exist a large number of patterns and/or long patterns. Frequent-pattern tree (FP-tree) structure, which is an extended prefix -tree structure for storing compressed, crucial information about frequent patterns, and develop an efficient FP-tree based mining method, FP-growth, for mining the complete set of frequent patterns by pattern fragment growth. Training dataset repeatedly produces massive amount of rules. It's very tough to store, retrieve, prune, and sort a huge number of rules proficiently before applying to a classifier. In such situation FP is the best choice but problem with this approach is that it generates redundant FP Tree. In this paper, the limitation of these two methods and an integrated techniques of both Apriori and FP-Growth, is used to overcome the limitation of existing methods.

Index Terms- Apriori, FP-Growth, Association Rule, Item Set

I. INTRODUCTION

Discovery of frequent item sets [1] is a very important data mining problem with numerous practical applications. Informally, frequent itemsets are subsets frequently occurring on a collection of sets of items. Frequent itemsets are typically used to generate association rules. However, from generation of rules is a rather straightforward task, the focus of researchers has been mostly on optimizing the frequent item set discovery step. Many frequent item set mining algorithms have been developed. The two most prominent classes of algorithms are Apriori-like and pattern-growth methods. Apriori- like solutions, represented by a classic Apriori algorithm [3], perform a breadth-first search of the pattern space.

Apriori starts with discovering frequent itemsets of size 1, and then iteratively generates candidates from previously found smaller frequent itemsets and counts their occurrences in a database scan. The problems identified with Apriori are:

(1) Multiple database scans, and (2) huge number of candidates generated for dense datasets and/or low frequency threshold (minimum support).

To address the limitations of Apriori-like methods, a pattern-growth, which consists in a depth-first search of the pattern space. Pattern-growth methods also build larger frequent sets from smaller ones, but instead of candidate generation and testing, they exploit the idea of database projections. Typically, pattern-growth methods start by transforming the original database into some complex data structure, preferably fitting in main memory. A classic example of the pattern-growth family of algorithms is FP-growth, which transforms a database into FP-tree stored in main memory using just two database scans, and then performs mining on that optimized FP-tree structure [2].

II. APRIORI YECHNIQUE

Apriori algorithm is used for frequent item set mining and association rule learning. The algorithm use a level-wise search, where k-itemsets (An item set which contains k items is known as k-item set) are used to explore (k+1) -itemsets, to my frequent itemsets from the transactional database for Boolean association rules. In this algorithm, frequent subsets are extended one item at a time and this step is known as candidate generation process. Then groups of candidates are tested against the data. To count candidate item sets efficiently, Apriori uses breadth-first search method and a hash tree structure. It identifies the frequent individual items in the database and extends them to larger and larger item sets as long as those item sets appear sufficiently often in the database. Apriori algorithm determines frequent item sets that can be used to determine association rules which highlight general trends in the database. The following is the procedure for Apriori algorithm:

C_k: Candidate item set of size k
L_k: frequent item set of size k
L_1 = {frequent items};
For (k = 1; L_k!=∅; k++)
do begin
C_{k+1} = candidates generated from L_k;
for each transaction t in database do increment the count of all candidates in C_{k+1} that are contained in t
L_{k+1} = candidates in C_{k+1} with min_support
end
return $\cup_k L_k$.
The following example explains Apriori technique.

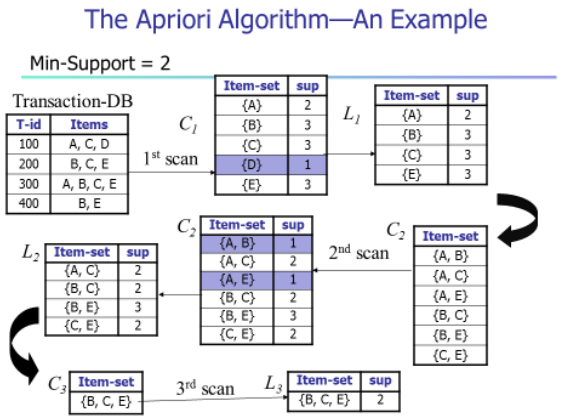

Figure 1.1 Apriori example

Limitation- the Apriori achieves good performance gained by reducing the size of candidate sets. However, in situations with a large number of frequent patterns, long patterns, or quite low minimum support thresholds, an Apriori-like algorithm may suffer from the following two nontrivial costs:

1) It is costly to handle a huge number of candidate sets. For example, if there are 104 frequent 1-itemsets, the Apriori algorithm will need to generate more than 107 length-2 candidates and accumulate and test their occurrence frequencies. Moreover, to discover a frequent pattern of size 100, such as $\{a_1,...,a_{100}\}$, it must generate $2100 - 2 \approx 1030$ candidates in total. This is the inherent cost of candidate generation, no matter what implementation technique is applied.

2) It is tedious to repeatedly scan the database and check a large set of candidates by pattern matching, which is especially true for mining long patterns [4].

III. FP-GROWTH TECHNIQUE

Let I ={a1,a2,...,am} be a set of items, and a transaction database DB(T1,T2……Ti(I belongs to (1…N))is a transaction which contains a set of items in I .The support 1 (or occurrence frequency) of a pattern A, where A is a set of items, is the number of transactions containing A in DB. A pattern A is frequent if A's support is no less than a pre defined minimum support threshold, ξ. Given a transaction database DB and a minimum support threshold ξ, the problem of finding the complete set of frequent patterns is called the frequent-pattern mining problem [3].

The FP- Growth algorithm for mining frequent patterns using FP-Tree by pattern fragment growth is: Input: a FP-Tree constructed with the algorithm mentioned in Algorithm for FP-tree construction [5].

D - Transaction database

ξ - Minimum support Threshold.

Output: The complete set of frequent patterns.

Method: Call FP-growth (FP-tree, null)

```
Procedure  FP-growth (Tree, A)
 {
 if (Tree contains a single path P)
then for each (combination (denoted as B) of the nodes in the path P)
do generate pattern B ∪ A with  support = minimum support of nodes in B;
else (for each a_i in the header of Tree)
do
            {
generate pattern B = ai ∪ A with support =a_i.support;
construct B's conditional pattern base and
                        then B's conditional FP-Tree TreeB;
 if (TreeB ≠ ∅)
            {

call FP-growth (TreeB, B)
  }
}
```

Above snippet written in **R** are essentially ephemeral, written for a single piece of data analysis.

Limitation- FP-Growth lack of good candidate generation method [1].

IV. IAFP TECHNIQUE

The main drawback of Apriori algorithm is that the candidate set generation is costly, particularly when massive number of patterns or long patterns subsist. The main drawback of FP-growth algorithm is the explosive quantity of lacks a good candidate generation method [6].

IAFP technique combines FP-Tree with Apriori candidate generation method to solve the disadvantages of both Apriori and FP-growth. The new algorithm will reduce the storage space, improves the efficiency and accuracy of the algorithm.

The outline of IAFP is given below:

- Initialization: Use an Apriori algorithm to mine all frequent patterns up to a small size.
- Iteration:
  - At each iteration, k seed patterns are randomly picked from the current pattern pool.
  - For each seed pattern thus picked, we find all the patterns within a bounding ball centered in the seed pattern.
  - All these patterns found are fused together to generate FP-tree and find a set of super-patterns. All the super-patterns, thus generated from a new pool for the next iteration.
- Termination: when the current pool contains no more than K patterns at the beginning of an iteration.

V. EXPERIMENTAL RESULTS

The proposed system is implemented using the Java language with a help of IDE called Eclipse and Jfreechart. jar file. Eclipse is an integrated development environment (IDE) for developing rich client applications. JfreeChart is an open source

library developed in Java, which can be used within Java based applications to create a wide range of charts.

The dataset considered for conducting analysis is given below.

ID	Transaction
1	Beer, Beans, Strawberries
2	Beans, Apple
3	Beer, Strawberries
4	Apple, Tomatos
5	Beans, Tomatos, Lemon, Computer
6	Apple, Lemon
7	Beans, Strawberries, Tomatos
8	Beans, Tomatos, Sugar
9	Apple, Tomatos, Sheets
10	Beer, Beans, Strawberries, Tomatos, Lemon
11	Beer, Beans, Lemon
12	Beer, Beans, Tomatos, Lemon, Sugar
13	Beer, Tomatos
14	Beer, Lemon
15	Beer, Beans, Strawberries, Apple, Tomatos, Lemon
16	Beans, Apple, Tomatos
17	Beer, Beans, Lemon, Sheets
18	Beans, Sugar
19	Apple, Sheets
20	Apple, Tomatos, Lemon, Sheets, Caviar

Figure 5.1 Dataset for Experiment

The following figure shows the comparison between Apriori, FP-Growth and IAFP algorithms.

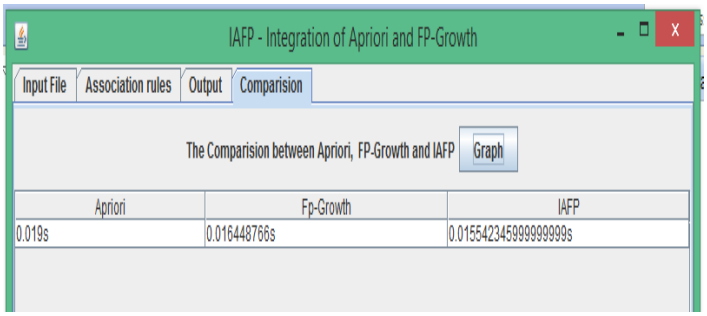

Figure 5.2 comparison between algorithms

The following figure shows comparison between Apriori, FP-Growth and IAFP algorithms in the form of a graph.

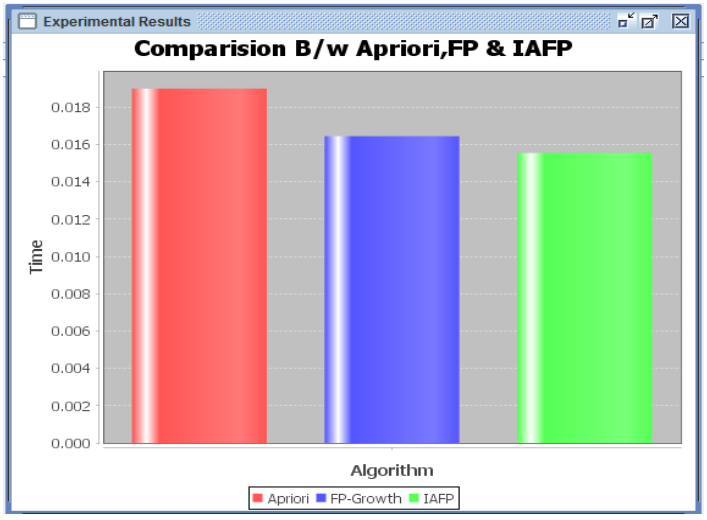

Figure 5.3 comparison between Apriori, FP-Growth &IAFP

VI. CONCLUSION

A new hybrid approach for data mining process is been proposed. Data mining is the current focus of research since last decade due to the enormous amount of data and information in modern day. The association is a topic of concern among various data mining techniques. This hybrid approach is to deal with large size data. Proposed system is the enhancement of both Apriori and Frequent pattern (FP) technique of association.

REFERENCES

[1] Jiawei Han, Micheline Kamber, ―Data mining concepts and techniques‖, Elsevier Inc., Second Edition, San Francisco, 2006

[2] Charalampos Vassiliou, Dimitrios Stamoulis, Anastasios, ―Creating Adaptive Web Sites Using Personalization Techniques: A Unified, Integrated Approach and the Role of Evaluation", Greece, Idea Group Publishing, 2003, pp. 261- 285,ch 12

[3] Jaideep Srivastava, Robert Cooleyz, Mukund Deshpande, Pang- Ning Tan proposed ―Web Usage Mining: Discovery and Applications of Usage Patterns from Web Datal, 2000.

[4] Yogita S. Pagar, Vishakha. R. Mote, Rahul S. Bramhane, ―Web Personalization using Web Mining Techniques", Emerging Trends in Computer Science and Information Technol2012 (ETCSIT2012)

[5] Liana Razmerita, Thierry Nabeth, Kathrin Kirchner, ‖User Modeling and Attention Support: Towards a Framework of Personalization Techniques", The Fifth International Conference on Advances in Human-oriented and Personalized Mechanisms, Technologies, and Services, 2012.

[6] Rahul Mishra, Abha Choubey, ―Comparative Analysis of Apriori Algorithm and Frequent Pattern Algorithm for Frequent Pattern Mining in Web Log Data", International Journal of Computer Science and Information Technologies, Vol. 3 (4) , 2012,4662 – 4665 P. Ronhovde and Z. Nussinov. Multiresolution community detection for megascale networks by information-based replica correlations. Physical Review E, 80(1):1–18, July 2009.

AUTHORS

First Author – Manasa G – Manasa G received the B.E degree from the Department of Information Science and Engineering at Visvesvaraya Technological Univesity, Belgaum and currently pursuing M.Tech degree in Computer Science and Engineering at Visvesvaraya Technological Univesity, Belgaum., Email-ID is: manasa9112@gmail.com

Second Author – Mrs. Kulakarni Varsha completed B.E & M.E in Computer Science & Engineering. Currently working as Assistant Professor in SVCE, Bangalore. Email-ID is: varsha_kulkarni@yahoo.com