Approaches for Mining Frequent Itemsets and Minimal Association Rules

Prajakta R. Tanksali
M.E Student
Department of Information Engineering
Padre Conceicao College of Engineering, Verna, Goa/Goa University

Abstract

Frequent itemsets mining is a popular and common concept used in day-to-day life in many application areas including Web usage mining, intrusion detection, and bioinformatics etc. This is the right place were the various Frequent Itemsets Mining Algorithms are used, which help the store manager/in-charge to arrange these items in a particular fashion so that the number of items purchased by the customers increase, thereby increasing the sales of the store. Such information can be used as the basis for decisions about marketing activities such as, promotional offers, seasonal offers or product placements. This paper presents a literature study of the different approaches to achieve the goal of frequent itemsets mining. We have tried to design an application for a chemist using these algorithms on a medical pharmacy dataset to help the shop owner maintain his stocks well and as per the user requirements.

Keywords- - Itemset, Frequent Itemset, Support Count/Threshold, Support, Confidence, Association Rules

I. INTRODUCTION

The overall goal of data mining process is to extract information from a data set and transform it into an understandable structure for further use. We need to select the best way of getting the items. Whenever we go to any shop, we get confused about what should be purchased because of large set of data items in the store's database. So the shopkeepers apply many algorithms for finding the best and efficient way of providing these products to the customer.

Data Mining encompasses tools and techniques for the 'extraction' or 'mining' of useful knowledge from large amount of data. This process is about finding patterns and relationships within the data items that can be possibly used for another purpose. One of the most important data mining applications is that of mining association rules. Association rules were first introduced by Agarwal. These are helpful for analyzing customer behavior in retail trade, banking system, web usage mining, bioinformatics etc. Association rule can be defined as {X,Y}=>{Z}, which means if customer buys X & Y, he is likely to also buy Z. Association rule mining process is to find out association rules among the items that satisfy the predefined minimum support and confidence from a given database. Therefore an item set is said to be frequent if it satisfies the minimum support and confidence.

II. FREQUENT ITEMSETS MINING APPROACHES

Frequent patterns such as frequent itemsets, substructures, sequences, phrase sets and sub graphs, generally exists in real world databases. Identifying frequent itemsets is one of the most important issues faced by the knowledge discovery and data mining community.

As frequent data itemsets mining are very important in mining the association rules. Therefore there are various techniques proposed for generating frequent itemsets. The algorithms used for this purpose vary in the generation of candidate itemsets and support count. These approaches for generating frequent itemsets are divided into basic three categories:

A. Horizontal Layout Based Data Mining Techniques

In horizontal layout, each row of database represents a transaction which has a transaction identifier (TID), followed by a set of items. In our research we have considered Apriori algorithm under horizontal layout category for experimental purpose.

B. Apriori Algorithm

This is the most classical and important algorithm for mining frequent itemsets proposed by Agarwal. Apriori uses breadth-first search to count candidate itemsets efficiently. It generates candidate itemsets of length (k+1) from itemsets of length k. Then it prunes the candidates which have an infrequent sub pattern.

Apriori is designed to operate on databases containing transactions (for example, collections of items bought by customers or details of a website frequentation).

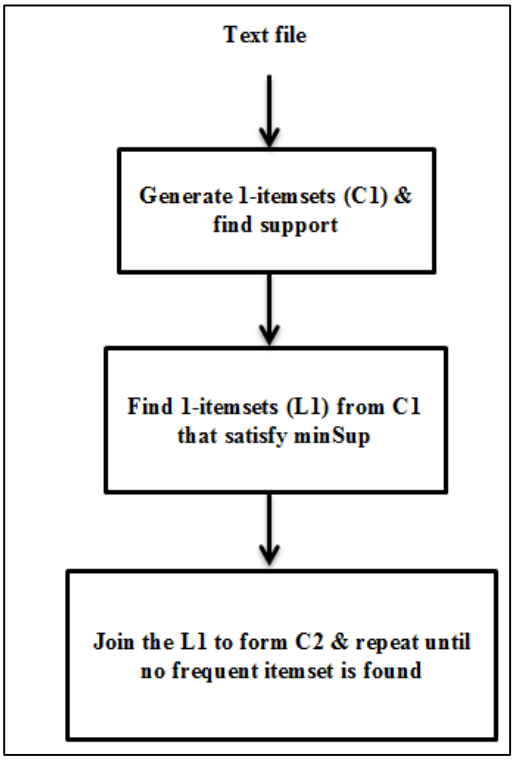

Fig. 1: Flow of Apriori algorithm

C. Vertical Layout Based Data Mining Techniques

In vertical layout**,** each column corresponds to an item followed by a TID list, which is the list of rows that the item appears. We have studied the Transaction Mapping algorithm here.

1) Transaction Mapping Algorithm

In this algorithm, transaction ids of each itemset are mapped and compressed to continuous transaction intervals in a different space and the counting of itemsets is performed by intersecting these interval lists in a depth-first order along the lexicographic tree. When the compression coefficient becomes smaller than the average number of comparisons for intervals intersection at a certain level, the algorithm switches to transaction id intersection. All the transactions that contain an item are represented with an Interval List. Each node in the transaction tree will be associated with an interval. The interval lists for each item is done recursively starting from the root in a depth-first order.

The TM algorithm requires construction of the Transaction tree which scans the database twice. The algorithm uses a lexicographic prefix tree data structure to generate the candidate itemsets.

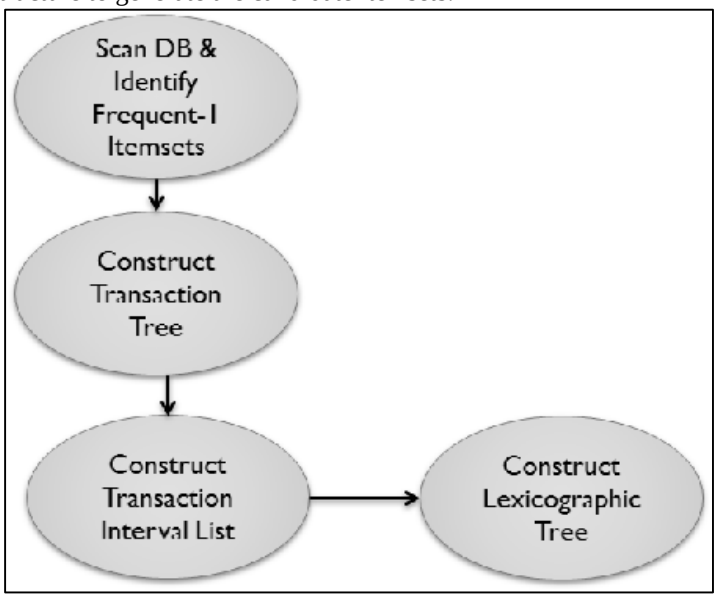

Fig. 2: Flow of TM algorithm

2) Lexicographic Prefix Tree Construction

Each node in the tree stores a collection of frequent itemsets together with the support count. The root contains all frequent 1-itemsets. Each edge in the tree is labelled with an item.

D. Projected Database Based Data Mining Techniques

In projected layout, we use tree structure to store and mine the itemsets. Here we have considered FP-growth algorithm.

1) FP-Growth Algorithm

It encodes the data set using a compact data structure called FP-tree and extracts frequent itemsets directly from this structure. This algorithm is based upon recursive divide and conquers strategy. First the set of frequent 1-itemset and their counts are discovered. Starting from each pattern, construct the conditional pattern base, then its conditional FP-tree is constructed (prefix tree). The items in each transaction are processed in L order, i.e. items in the set were sorted based on their frequencies in the descending order to form a list.

A FP-tree structure consists of, one root labelled as "null"; each other node having item-name, count, node-link; frequent-item header having item-name, head of node-link.

The size of the FP-tree depends on how the items are ordered. Ordering by decreasing support is typically used but it does not always lead to the smallest tree.

It has only two passes over the dataset. Also compresses a large database into a more compact tree structure i.e. FP-tree. Here there is no candidate generation and therefore no repeated database scans, that makes FP-growth faster than other algorithms.

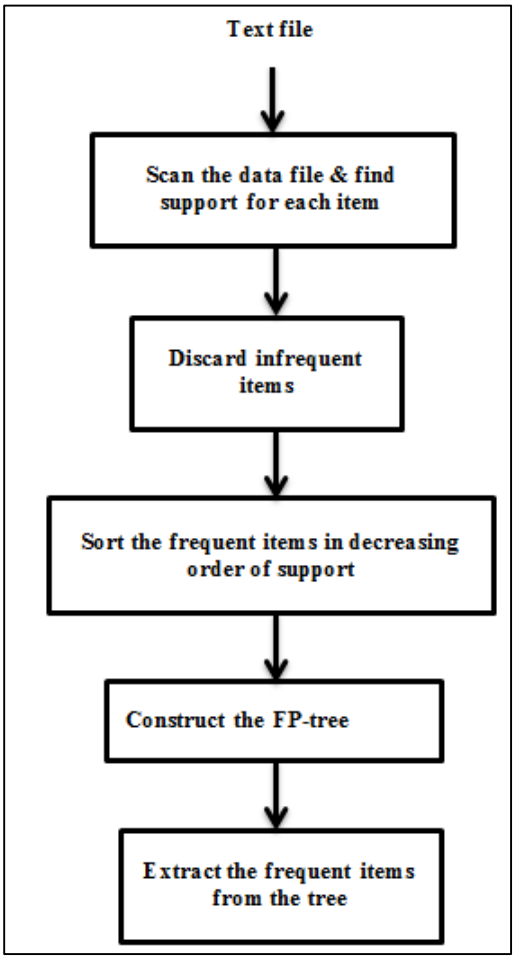

Fig. 3: Flow of FP-growth algorithm

III. EXPERIMENTAL RESULTS

In our research we have implemented two important frequent itemsets mining algorithms i.e., Apriori and FP-growth, for experimental purpose. We have used two datasets for analysis. One is the online fetched "grocery.txt" file, consisting transactions with various supermarket items purchased by customers. This dataset was used to get results at early stages of analysis.

Another dataset used was the "medicine.txt" file that was manually created based on the customer bill receipts collected from the medical store. This input transaction file is a text file with string values representing medicine names in each transaction. Total numbers of transactions are 2271 with 300 unique names. Certain items (medicines, cosmetics) in the dataset are very frequently bought (items with highest support count) whereas some are infrequent (items with low support count).
With a small size sample of the medicine dataset, both the algorithms gave almost similar results.

A. Apriori

```
Input configuration: 50 transactions
Min_Sup_Thres = 0.4%
[7] (0.66 33)
Found 1 frequent itemsets of size 1 (with support 40.0%)
Creating itemsets of size 2 based on 1 itemsets of size 1
Created 0 unique itemsets of size 2
Execution time is: 0.0 seconds.
Found 1 frequents sets for support 40.0% (absolute 20)
Done


Result: glycomet_500_tab -->| 7
```

Fig. 4: Output File-Apriori

B. FP-growth

```
Item:glycomet_500_tab Freq.:33
```

Fig. 5: Output File-FP-growth

IV. CONCLUSIONS

Using the above chemist dataset the following was concluded:
Item with max occurrence: betaloc_25mg_tab freq: 91
Item with least occurrence: naturolax_powder_300gm freq: 1
Although Apriori algorithm has certain limitations, it worked well for medicine dataset used for the experiment.

Further, if the above medicine dataset is added with certain transactions with repeated individual single item transactions, Apriori has to simply scan the dataset repeatedly increasing the processing time and CPU utilization.

Deciding the Threshold value is the challenge! It is based on the domain of the data used and also on the size of the dataset. Most of the datasets used with frequent itemsets mining algorithms are raw experimental data in integer or float format. Therefore finding frequent itemsets from the dataset and applying evaluating measures like minimum support threshold, minimum confidence, mean, standard variance etc. is simple.
To use Apriori algorithm more efficiently

- Scan the transactions to determine the support of each candidate itemset.
- To reduce the number of comparisons, store the candidates in a hash structure i.e. Instead of matching each transaction against every candidate, match it against candidates present in the hashed buckets.

FP-growth is better method than Apriori, as it scans the database only twice and thus making process faster.

V. FUTURE SCOPE

Similar chemist data from different locations in a city could be taken into consideration to study the pattern of medicines bought and thereby conclude any particular diseases in a locality. Study of different diseases based on the changing medicine trends could further prevent any epidemic in the society.

For a chemist, this application could maintain a record of the stocks and discard infrequent or rarely bought items. Study of various issues with FP-growth algorithm like tree construction and processing time could be done. Also other frequent itemsets mining algorithms could be studied and implemented for better results.

ACKNOWLEDGMENT

The success of this paper depends largely on the encouragement and guidelines of many people. I therefore would like to extend my sincere gratitude to all of them who made it possible for me to complete my research successfully.
I wish to express my sincere thanks to Dr. Luis C. Mesquita, Principal of Padre Conceicao College of Engineering, for granting me permission to carry on with my research work.

I am very much grateful to my guide Mr. Ameya Wadekar (Assistant Professor) for stimulating supervision, continuous guidance, suggestions whenever required during my dissertation work.

I also wish to express my sincere thanks to Mr. Ranjeet Sardesai (Navajivan Medical Stores, Vasco) and Mr. Swapnil Gawde (Saraswati Medical Stores, Vasco) for providing the required data for my experiments.

REFERENCES

[1] Mingjun Song and Sanguthevar Rajasekaran, "A Transaction Mapping Algorithm for Frequent Itemsets Mining", IEEE Transactions On Knowledge And Data Engineering, VOL. 18, NO. 4, APRIL 2006.
[2] Christian Borgelt,"An Implementation of the FP-growth Algorithm".
[3] Jiawei Han und Micheline Kamber, "Frequent Item set Mining Methods".
[4] S. Neelima, N. Satyanarayana and P. Krishna Murthy, "A Survey on Approaches for Mining Frequent Itemsets".
[5] Pratima Gautam, Dr. K. R. Pardasani, "Algorithm for Efficient Multilevel Association Rule Mining", (IJCSE) International Journal on Computer Science and Engineering Vol. 02, No. 05, 2010, 1700-1704.
[6] Ya-Han Hu, Fan Wu, Yi-Jiun Liao, 'An efficient tree-based algorithm for mining sequential patterns with multiple minimum supports', The Journal of Systems and Software 86 (2013) 1224–1238.
[7] Ke-Chung Lin, I-En Liao, Tsui-Ping Chang, Shu-Fan Lin, "A frequent itemset mining algorithm based on the Principle of Inclusion–Exclusion and transaction mapping".
[8] Trieu Anh Tuan Ritsumeikan University 2012, "A Vertical Representation for Parallel dEclat Algorithm in Frequent Itemset Mining".
[9] Paresh Tanna, Dr. Yogesh Ghodasara, "Using Apriori with WEKA for Frequent Pattern Mining".
[10] Kanu Patel, Vatsal Shah, Jitendra Patel, Jayna Donga, "Comparison of Various Association Rule Mining Algorithm on Frequent Itemsets".