IJCTA, 9(27), 2016, pp. 367-373 ISSN: 0974-5572
© International Science Press

A Study on Frequent Pattern Mining

Ramah Sivakumar* **and J.G.R. Sathiaseelan****

ABSTRACT

Pattern mining plays a vital role in all data mining applications. Each data can be considered as a pattern. Data may include text, image, audio, video, etc., all may be considered as patterns. In specific, pattern which occurs frequently that is again and again is referred to as frequent pattern. In transactional databases the dataflow is huge, and by mining the frequent patterns meaningful information could be found. Rules can be framed based on the association of data. These rules are termed as association rules. Frequent patterns are also useful for various datamining problems such as classification and clustering. Other than this there are enumerable applications of frequent patterns in various fields which include bioinformatics, spatiotemporal data, social network analysis, business intelligence and software bug detection. Researches based on algorithms for frequent pattern mining have also been still in exploration. This paper provides knowledge about mining frequent patterns.

Keywords: Pattern mining, frequent patterns, association rules

1. INTRODUCTION

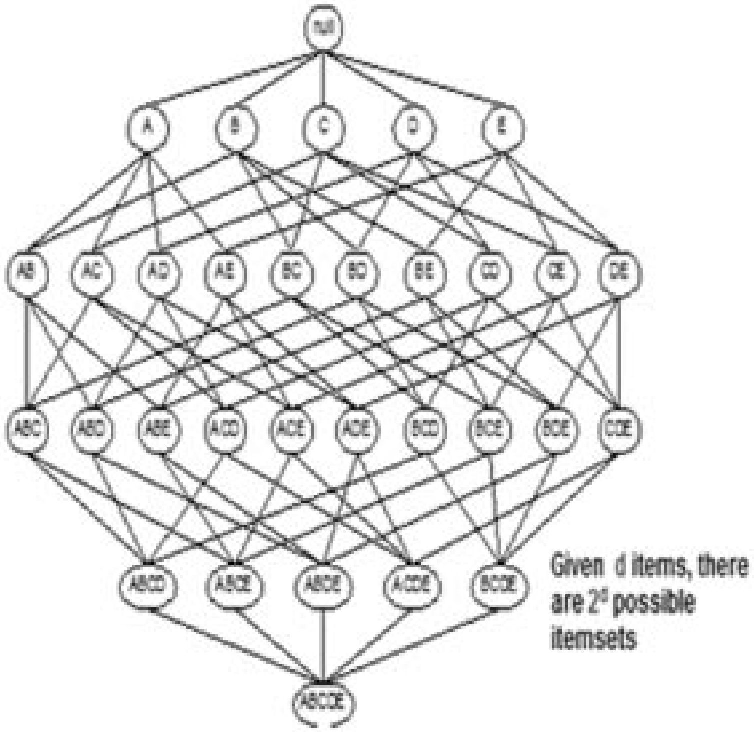
The main reason for mining frequent patterns is to find relationship among data in the database. The problem can be stated as follows:

> *Given a database DB with transactions such as T1, T2……,T_N , find out pattern P which is present in at least a fraction r of the transactions.*

Here the fraction r is referred to as the minimum support. This measure r can be represented either as a whole number or as a fraction. That is number of occurrences of the pattern divided by the total number of transactions. For example, if a pattern P appears in 3 transactions and the total number of transactions is 10 then the support of the pattern P is 3/10. Value of r is 3/10 for that particular pattern P. The problem of mining frequent patterns was initially proposed for business transactions that are for market basket data [1]. The purpose was to find the frequently bought items, and the items which are bought together. Based on the buying behaviour of the customer, the shelf life of the itemsets were modified which in turn improves the business profit. Customer retention, fraud detection, better customer relations could also be achieved. Presently, this approach have found new applications such as software bug analysis, web log mining. Frequent patterns may be frequent itemsets, subsequences, or substructures that occur frequently in a dataset. Set of items such as bread and jam that occur frequently is referred to as frequent itemset. A subsequence can be referred to as buying a PC and after that a memory card and then a printer subsequently and if it occurs frequently in a transactional database then they are frequent subsequence[2]. Subgraphs, subtrees, sublattices which combine with frequent itemsets or subsequences can be referred to as frequent substructures. Based on the threshold value that is support fixed by the user, frequent patterns are identified. In the next step all those patterns whose support is less than the defined which are termed as infrequent patterns are removed from further processing. Then the confidence level is checked to determine the association among the frequent patterns. This analysis of transactional data gives details about current processing in the database which is also cheaper considered to other manual methods.

* Assistant Professor Department of Computer Science Bishop Heber College Trichy-17, *Email: rmhsvkmr@yahoo.co.in*

** Associate Professor and Head Department of Computer Science Bishop Heber College Trichy-17, *Email: jgrsathiseelan@yahoo.com*



Given d items, there are 2^d possible itemsets

Figure 1: Number of possible itemset for given items

Originally in the initial model, frequent patterns were mined to frame association rules. The output of mining frequent patterns is framing association rules and it is considered to be the second stage of the problem. These association rules are derived from the frequent patterns. Association rules[3,8] play an important role in several areas such as telecommunication networks, market and risk management, inventory control etc., Confidence which is a measure plays a vital role in mining association rules. If there exists items A and B in a transaction database, then A=>B is considered as association rule at minimum support *r* and minimum confidence *c* based on the following two conditions.

The set A U B is a frequent pattern

The ratio of the support of A U B to that of A is atleast c

2. DEFINITIONS

Pattern: It is an attribute of the database. For example, in market basket it may be bread, butter, milk etc.,

Itemset: Set of items in a transaction. That is collection of attributes of the database.

Transaction: It means a record in the database. Ti = $\{i_1, i_2, \ldots, i_n\}$. Each transaction is identified using an identifier called transaction ID (Tid). The database DB constitute whole set of transactions Ti. DB ={T1, T2, …, Tn}

Support Count: Frequency of occurrence of a particular pattern.

Support: The support of A=> B in transaction database is a ratio. The ratio is between the number of item sets that contain A and B, and the total number of item sets. That marks support (A=>B). That is the percentage of item sets containing A and B in the transaction database.

Confidence: It is the ratio between number of transactions containing A and B and the number of transactions containing A. That is marked as confidence (A =>B).

Frequent pattern: The pattern, whose support is greater than or equal to the pre-defined minimum support (Min Sup).

3. FREQUENT PATTERN MINING ALGORITHMS

Algorithms for frequent pattern mining are based on support-confidence framework. In this framework patterns above the given threshold value are found out. Here when the support value is low, the search space needed for frequent patterns is enormous. Therefore, it becomes a tedious one for analysis of patterns in less space and time. So the focus was to develop an algorithm which provides an optimised computational efficiency.

Some specialized frameworks have also been designed which mines more interesting patterns. Different measures are used in these specialized frameworks.

The algorithm given below is "baseline" algorithm which really forms the baseline for most of the frequent pattern mining algorithms.

A basic frequent pattern mining algorithm

```
Algorithm Baseline mining (Database: T, Minimum support s)
begin
FP = {}
Insert length-one frequent pattern in FP
until all frequent patterns in FP are explored do
begin
generate a candidate pattern P from one (or more) frequent pattern(s) in FP
if support (P, T) > = s
Add P to frequent pattern set FP
end
end
```

In the above algorithm T is transactional database and s is the support value. All length one frequent patterns are populated first and stored in *FP* which is the pattern data store. Then candidate patterns are generated and the support values are compared in the database. The patterns whose support value are higher than or equal to the specified support value are stored in *FP*. Until all the frequent patterns are found out from the database, this process is continued. Joins are used in the candidate generation process. Join based algorithms generate (k + 1) candidates from frequent k patterns with the use of joins.

Join based algorithms:

1. Apriori
2. Apriori T*id*
3. Apriori
4. DHP algorithm

Apriori algorithm

```
Algorithm Apriori (Database: T, support: s)
Begin
Generate frequent-1 patterns and 2-patterns
using specialized counting methods and
denote by $F_1$ $F_2$;
k:=2;
```

```
while $F_k$ is not empty do
begin
generate $C_{k+1}$by using joins on $F_k$
prune $C_{k+1}$ with *Apriori* subset pruning trick;
generate $F_{k+1}$by counting candidates in
$C_{k+1}$ with respect to T at support s;
k: = k + 1;
end
return $U^k_{i=1}$ $F_i$;
end
```

Apriori algorithm which is a classical one is based on this category [10]. This is also referred to as level-wise exploration. Here, the main property of Apriori states that the subset of every frequent pattern is also frequent. The main steps in Apriori include candidate generation and testing, join step and prune step. This concept is very useful as it provides an understanding of how the search space of candidate patterns may be explored in order and non-redundant way.

Apriori like methods[13,16] were developed by many number of researchers in the later periods such as Apriori Tid and Apriori Hybrid[13,17]. As the enumeration tree concept provides more flexible framework these methods also use the same concept for mining frequent patterns [18].

Direct Hashing and Pruning also known as DHP algorithm [3], was proposed after Apriori algorithm in which two main optimization were introduced to Apriori method. They include pruning the candidate itemsets in each iteration, and the second one was to trim the transactions by which support counting process was made more efficient.

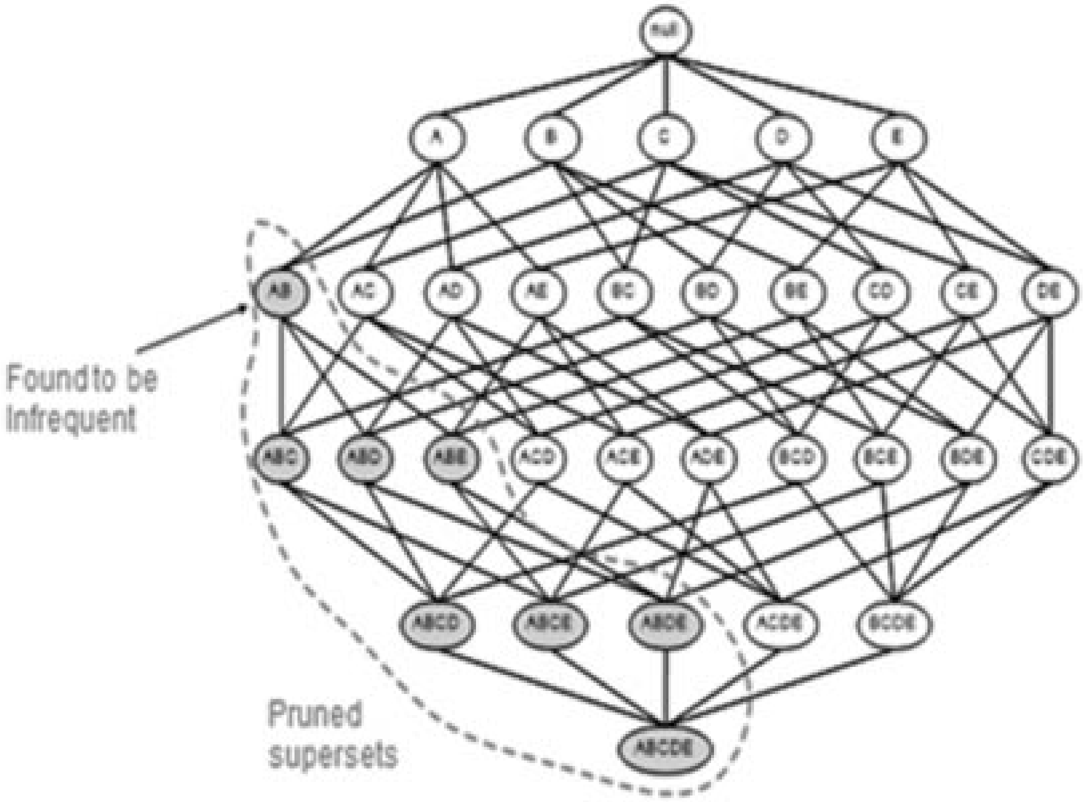


Figure: 2 Pruning Process in Apriori Algorithm

Tree based algorithms

AIS Algorithm

Tree projection Algorithm

Vertical mining algorithm

Eclat

Viper

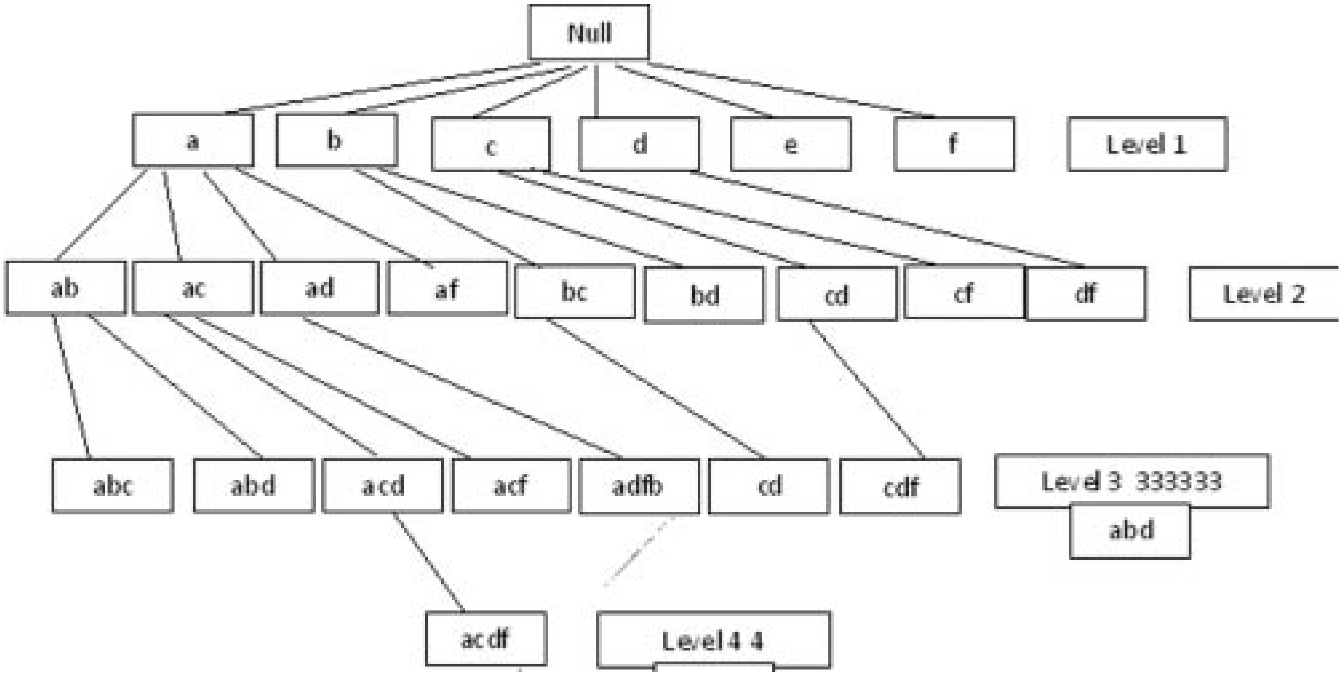


Figure 3: The lexicographic tree

Set enumeration concepts form the base of tree based algorithms. Using the subgraph from the lattice of itemsets, the candidates are discovered which is referred to as lexicographic tree or enumeration tree[5]. Breadth first or depth first order is used to build the tree, which provides a specific order of exploration that is used in many situations.

For efficient counting of patterns the vertical pattern mining algorithms employs vertical representation of the transaction database[4]. A *tidset* or *tidlist* is formed by which the support of k-patterns is calculated by intersection of the *tidlist*. The tidlist contains the set of items in each transaction.

Table 1
vertical representation of transactions

Pattern	*Tidlist*
A	1,2,5
B	1,2,3,4
C	1,2,4
D	1,3,5
E	1,2,3
F	4,5
G	3,5

Eclat algorithm uses breadth first approach. The depth first version of *Eclat is dEclat* [12]. *dEclat* uses recursive Tidlist intersection with differencing. Viper algorithm [4] also uses vertical approach for mining. The vertical database is represented in the form of compressed bit vectors. These are also mentioned as *snakes*. To achieve efficient counting of frequent patterns these *snakes* are used. Viper differs from Eclat only by the compressed bit vector representation.

4. SCALABILITY PROBLEMS IN FREQUENT PATTERN MINING

In data streams

With Big data

Due to various advances in hardware and software data streams have become very popular. For these types of scenarios, mining algorithms have to be executed in single pass. As frequent and sequential pattern mining methods use level wise methods this becomes a challenging one. Two alternatives for frequent pattern mining in data streams include

Frequent items or *heavy hitters*[6]

Frequent itemsets

While mining frequent patterns with big data access cost is the major issue when data have already been stored in distributed frameworks. Big data algorithms for mining frequent patterns are based on *MapReduce [7,15]* framework.

5. APPLICATIONS OF FREQUENT PATTERN MINING

Applications can be classified into two types:

Application in data mining problems such as classification, clustering, and outlier detection.

In common applications such as web mining, chemical and bioinformatics, software bug analysis.

Some of the applications are discussed here under:

Market basket analysis: Every transaction would be comprised of set of items based on the customer buying behaviour [3]. Based on the frequent patterns customer buying behaviour could be analysed. Based on the association of items association rules could be framed. Decision making such as shelf life of items, stock taking and recommendations based on customers could be made using frequent patterns which in turn improves the business profit.

Chemical and biological analysis: Frequent patterns play a vital role in these domains as the data are represented as graphs and sequences.

*Web log analytics: B*rowsing behaviour could be analysed based on the browsing behaviour. The results find importance in website design, outlier detections, and recommendations based on web.

Data mining problems: Frequent patterns could be used for problems such as classification, clustering, outlier analysis.

6. CONCLUSION

The main goal of frequent pattern mining is to obtain patterns and structures which frequently occur in a database. The database may be static or stream where there are huge amount of data flow. Patterns may indicate business, scientific, economic or social trends, and security threats. Based on the mined frequent patterns, association rules could be framed which is the second output stage of the process. Classical algorithms for mining frequent patterns used so many passes on the databases with no restriction. However, in data streams the passes are limited to single. Data window[14] based techniques are used in streams to mine frequent patterns. This paper provides an overview of frequent pattern mining. Efficient algorithms have been proposed for mining long patterns, interesting patterns, constrains based pattern mining and compression[11]. The latest issue in frequent pattern mining is scalability as enormous data has been created in numerous applications.

REFERENCES

[1] R. Agrawal, T.Imielinski, and Aswami, "Database Mining: A performance Perspective, IEEE Transactions on knowledge and Data engineering, 5(6), pp 914-925, 1993

[2] R. Agrawal and R.Srikkant, Mining Sequential Patterns, ICDE conference, 1995

[3] J.S. Park, MS.Chen, PS.Yu, "An effective Hash based Algorithm for mining Association Rules", ACM SIGMOD Conference 1995

[4] Shenoy, P. and Haritsa, J. and Sudarshan, S. and Bhalotia, G. and Bawa, M. and Shah, D., "VIPER: A Vertical Approach to Mining Association Rules", ACM SIGMOD International Conference on Management of Data (SIGMOD 2000), May 16-18, 2000 , Dallas, Texas.

[5] F.Geerts, B.Geothals, J.Bussche, "A tight Bound on the number of candidate Patterns", ICDM, Conference, 2001.

[6] Graham Cormode, Flip Korn, S. Muthukrishnan, Divesh Srivastava, " Finding Hierarchical Heavy Hitters in Data Streams", Proceedings of the 29th VLDB Conference, Berlin, Germany, 2003

[7] J. Dean, S.Ghemawat, "MapReduce: Simplified Data Processing on Large Clusters, OSDI, pp. 137-150, 2004

[8] J. Blanchard F.Guillet, R.Gras and H.Briand, "Using information-theoretic Measures to assess association rule Interestingness. ICDM Conference, 2005.

[9] A. Tiwari, R.K. Gupta and D.P. Agrawal, " A Survey on Frequent Pattern Mining: Current Status and Challenging Issues", Information Technology Journal Year: 2010 | Volume: 9 | Issue: 7 | Page No: 1278-1293 DOI: 10.3923/itj.2010.1278.1293

[10] Mamta Dhanda, Sonali Guglani , Gaurav Gupta, "Mining Efficient Association Rules Through Apriori Algorithm Using Attributes", In: International Journal of Computer Science and Technology Vol 2, Issue 3, September 2011, ISSN:0976-8491

[11] J Vreeken, M.Van Leeuwen and ASiebes, Krimp, " Mining itemsets that compress", Data Mining and knowledge Discovery, 23(1) pp169-214, 2011

[12] Tuan A. Trieu, Yoshitoshi Kunieda, " An improvement for dEclat algorithm", Proceeding ICUIMC '12 Proceedings of the 6th International Conference on Ubiquitous Information Management and Communication Article No. 54 ACM New York, NY, USA ©2012 ISBN: 978-1-4503-1172-4 doi:10.1145/2184751.2184818

[13] Jiao Yabing, "Research of an improved Apriori Algorithm in Data Mining Association Rules", International Journal of computer and communication Engineering, vol2, No1, January 2013

[14] Gangin Leea, Unil Yuna, Keun Ho Ryu, "Sliding window based weighted maximal frequent pattern mining over data streams", Expert Systems with Applications, Volume 41, Issue 2, 1 February 2014, Pages 694–708

[15] Dr. Siddaraju,Sowmya C L2, Rashmi K3, Rahul M, "Efficient Analysis of Big Data Using Map Reduce Framework", International Journal of Recent Development in Engineering and Technology (ISSN 2347-6435(Online) Volume 2, Issue 6, June 2014)

[16] Xiang Cheng, Sen Su, Shengzhi Xu, Zhengyi Li, "DP-Apriori: A differentially private frequent itemset mining algorithm based on transaction splitting", Computers & Security, volume 50, May 2015, pages 74-90

[17] Kawuu W. Lin, Sheng-Hao Chung, "A fast and resource efficient mining algorithm for discovering frequent patterns in distributed computing environments", Future Generation Computer Systems, Volume 52, November 2015, Pages 49-58

[18] Akshita Bhandari, Ashutosh Gupta, Debasis Das, "Improvised Apriori Algorithm using frequent pattern tree for real time applications in Data Mining", International Conference on Information and Communication Technologies Procedia computer Science, volume 46, 2015, pages 644-651

[19] Aya Hellal, Lotfi Ben Romdhane, "Minimal contrast frequent pattern mining for malware detection", Computers & Security, Volume 62, September 2016, Pages 19–32.