IJCSE International Journal of Computer Sciences and Engineering Open Access

Survey Paper Vol.-6, Issue-5, May 2018 E-ISSN: 2347-2693

A Survey and comparative study of the various algorithms for Frequent Itemset Mining

Uma.N[1*], Prashanth C.S.R[2],

[1]Dept. CSE, New Horizon College of Engineering, VTU, Bangalore, India
[2] Dept. CSE, New Horizon College of Engineering, VTU, Bangalore, India

Corresponding Author: umamam@gmail.com, Tel.:09986905934

Available online at: www.ijcseonline.org

Accepted: 21/May/2018, Published: 21/May/20182018

Abstract— In Data mining field, frequent item set mining is one of the most intensively investigated problems in terms of computational complexity. The concept is widely used in market basket analysis, finance, and health care systems. Finding frequent patterns plays an essential role in mining associations, correlations and much other interesting relationship among data. The interest in the problem still persists despite of elaborate research conducted in the last two decades, due to its computational complexity and the fact that the results sets can be exponentially large. This combinatorial explosion of frequent item set methods become even more problematic when they are applied to Big Data. In this survey paper, an effort is made to present various popular algorithms and its analysis.

Keywords—Frequent item set mining, Association rule mining, Big Data

I. INTRODUCTION

The recent rapid advancements in technology and science, changes in human life styles and the competition and business challenges, have given rise to production of massive amounts of data. This has lead to storage and computational challenges. The intelligent analysis of data is quite challenging, but can provide various useful business insights.

Association rule mining helps in discovering association rules, which helps in the process of decision making. The two stages in association rule mining are i)to find all frequent item sets and ii) to generate reliable association rules from all the frequent item sets. The two significant measures which is frequently used for association rules are i) support and ii)confidence.

Support(s) is an indication of the frequency with which an item occurs in a data set. For any data item X, in a transactional database T, with t transactions,

$$\text{Support(X)} = \frac{\text{Number of times X occurs in the transaction}}{\text{Total Number of Transactions}}$$

Confidence(c) is an indication of the frequency with which a rule is found to be true. The confidence value of a rule X=>Y with respect to transaction T is defined as proportion of transactions with the condition that those transactions which contain X also contains Y.

Confidence (XY) =Support (XUY) / Support(X)

It is a measure of degree of strength of association rules. The users can pre-define thresholds of support and confidence to drop the rules which are not so useful. The two thresholds are named **minimal support** and **minimal confidence**.

A set of items is referred to as **itemset**. Since the database is huge, the users will be interested only in the frequent itemsets. Four key challenges in association rule mining are

i) Determination of candidate frequent itemsets
ii) Determination of the support count of itemsets
iii) How to reduce the number of candidate itemsets
iv) How to reduce the associated database scans.

In this paper, an effort is made to understand the working of various frequent itemset mining algorithms and a comparative study is done. In section 2 of the paper related work is presented and in section 3, a study on the methods to improve the frequent item set mining algorithms are presented. The results of the discussion are presented in section 4 .The concluding section 5 summarizes the paper.

II. RELATED WORK

A. ASSOCIATION RULE MINING ALGORITHMS

Association rule mining is widely used technique for decision making, and to derive the useful business insights. The first stage in any association rule mining algorithm is to find all frequent item sets and the second stage being generation of various reliable association rules. Finding the computational complexity of frequency based problems such as frequent itemset mining pose a huge challenge. Most of the association rule mining algorithms adopt a divide and conquer strategy. Apriori[1], Eclat[2], and FP-Growth[3] are among the most common algorithms for frequent itemset mining.The two widely used basic approaches in representing the conditional database in frequent itemset mining are horizontal and vertical representations.

In a horizontal representation, the database is stored as a list of transactions, each of which is a list of the items contained in it. In a vertical representation, a database is represented by first referring with a list to the different items. For each item a list (or array) of identifiers is stored, which indicate the transactions that contain the item. The apriori algorithm and Split and Merge algorithm (SaM)[4] uses the horizontal representation, whereas Eclat algorithm and Viper algorithm[5] uses the vertical representation.

Divide and conquer strategy is widely adopted by various Frequent Itemset (FIS)mining algorithms. In divide and conquer approachor some chosen item X, the problem to find all frequent itemsets is split into two sub problems:

(i) Find all frequent item sets containing the item X and

(ii) Find all frequent item sets not containing the item X.

Each sub problem is then further divided based on another item y, based on whether y occurs with x(x,y), or not (x,!y).The other possible sub problems are (!x,y)and (!x,!y). All sub problems that occur in this divide-and-conquer recursion can be defined by a conditional transaction database and a prefix. The prefix is a set of items that has to be added to all frequent item sets that are discovered in the conditional database.

B.FREQUENT ITEMSET MINING ALGORITHMS

Almost two decades considerable research has been performed to compare the performance of the three prominent algorithms: Apriori, FP growth, Ectlat for evaluating the scalability of algorithms.

i)Apriori Algorithm

The name of algorithm is based on fact that it uses a downward closure property called Apriori among k frequent itemsets.A k itemset is frequent if and only if all its sub itemsets are frequent. Apriori uses generate & test approach. It first finds the 1 item frequent set and uses the support count information to find 2 item sets and so on.This process is continued until no more frequent item set generation is possible. The candidate itemsets in each phase whose support count are less thn the minimum threshold are pruned. Apriori is a very prominent algorithm for mining frequent itemsets for Boolean association rules. The main disadvantage of Apriori algorithm is that support counting is expensive while performing pattern checking .It also has to do multiple database scans and also generates a huge number of candidates even after applying the apriori principle.

ii)FP-Growth

FP-Growth is an algorithm which can mine the complete frequent itemset without candidate generation. It employs a divide and conquer strategy and can derive all frequent itemsets in two scans. The first scan of the database derives a list of frequent items in which items are ordered by a descending frequency order. The items that do not satisfy a given minimum support, they are pruned according to Anti-monotone property. The List is then compressed into a frequent pattern tree or FP-Tree.FP-Tree is constructed as follows: First we create the root of the tree and label it as NULL. After removing the infrequent items, the database is scanned again, and the sorted transactions are inserted into a global FP-tree starting from the item with the lowest support. After inserting all of the transactions in the same way, FP-tree is completed. After the tree construction; FP-growth performs mining operations selecting the bottom item in the header table first. The algorithm then confirms all of the node locations for the item through the link-node information, and searches the tree from the nodes with the item to tree’s root, where the searched paths become a conditional database. After that, using the conditional database, FP-growth generates a conditional FP-tree containing the selected item ‘i’ (i.e. i’s conditional FP-tree). In the tree, one item is selected again and the algorithm iterates expansion steps until a single-path is generated. If a single-path is discovered, the expansion is stopped and the algorithm extracts frequent patterns combining the items selected so far with all of the items in the current conditional FP-tree. FP-growth algorithm is efficient and scalable and is faster than Apriori algorithm

iii)Eclat Algorithm

Eclat is basically a depth-first search algorithm which uses a vertical database representation. It exploits the set intersection method. It partitions the item sets and at most uses three scans of local database. When the database is represented using vertical layout, support counting is done in an easy and efficient manner.

Some of the factors which we can use to compare the performance of these algorithms are

i)The effect of representation of conditional databases.

ii)The effect of Dataset characteristics

iii)Candidate Generation &Pruning Strategy

iv)Number of Database scans required

III. METHODS TO IMPROVE EFFICIENCY OF FIM ALGORITHMS

Various algorithms were implemented to overcome the drawbacks of the above discussed algorithms. Below presented are some improvisation done for the Apriori, FP-Growth and Eclat.

i)Improvisations on Apriori Algorithm[5][6][7]

a) Hash-Based Techniques: In this method hashing technique is used and item sets are put in the corresponding buckets. Each Transaction in the database is scanned and is hashed to the different buckets thereby keeping track of table count.

b)Transaction Reduction: In this method we use the some transaction reduction technique in which the number of transactions which will be scanned for future iterations are reduced.

c) Partitioning: In this method we partition the complete data set into smaller sets to find the candidate items. Each partition will be used to generate locally frequent item sets, and then these local frequent item sets are regarded as candidate local frequent item sets are used to get the final global frequent item sets through testing their support.

d) Sampling: In this technique, we choose only a subset of data to work on which gives results in less time. In this method, but some degree of accuracy is traded off against efficiency.

e)Dynamic Itemset Counting: In this method we add the item sets at different points during scanning a whole data set.

Various research are conducted about how to improve the efficiency of the apriori algorithm. One of the improvement was based on the set operation which could solve the two key problems of reducing the times of scanning the transactional database and reducing the number of candidate item sets[8].FUP(Fast Update)[9] was proposed, for the incremental mining association rules. When new transactions are added to the database, the FUP algorithm updates the association rules in a database. Algorithm FUP is used the concept of Apriori and is designed to discover the new frequent itemsets iteratively.

ii) Improvisations on FP-Growth Algorithm

FP-Growth Algorithm does not generate any candidate sets.But it has several shortcomings. Some of the shortcomings specific to FP-Growth algorithm are

i) FP-Growth algorithm could generate too many branches, waste memory resources, prolong the time of recursive search and reduce the efficiency of FP-tree mining if the comparison with the suffix node is not done.

ii) The FP-Tree scanning costs time and the time component consists of 2 parts:FP-tree building time and the traversing the FP-tree to mine the frequent item sets.

iii) The conditional FP-Tree is stored in another space .The item is forward mapping whereas conditional FP pattern base is done in backward mapping and can result in conflicts.

The ENFP (Exchange Node FP) growth algorithm[10],addresses these shortcomings by improvising on the FP-Tree structure, with a node switching strategy, and with the help of FP array Auxiliary with backward mapping. To overcome the drawback of FP-Growth algorithm which can handle efficiently only small data sets and whose performance degrades with the large data sets, various research were conducted. One of the improvements was to sample the large data set using systematic sampling technique and make a SOM(Self Organizing Map) cluster analysis[11].Then, the large data set is partitioned into several subsets according to the cluster results. The FP-growth algorithm is executed in each subset, and association rules are mined.

To overcome the drawback due to the FP tree generation and conditional pattern base, an improvisation was done based on compound single linked list [12].In this paper the improvisation is done by using the sequencing table and single linked list as the main data structure. Further it does not generate conditional FP-tree. The mining is done only in one direction, using the header table in the FP-tree, which is stored in a sequence table, ordered in the descending sequence according to the minimum support threshold min_sup, and then a compound single linked list is formed. Through traversing each transaction’s frequent item sets stored in its single linked list, mining of the frequent patterns directly without generating conditional Fptree is possible.

Painting-Growth algorithm and N Painting-Growth algorithm[13]algorithms get all frequent item sets only through the two-item permutation sets of transactions, and only by scanning the database once.

To extend the FIM in data streams, and to address the time complexity various strategies like sliding window mechanisms etc are adopted [14].

iii) Improvisations on Eclat Algorithm

Eclat algorithm suffers from the following short comings:
i) Practically, Tid sets may be quite long. This can increase the memory space as well as much computation time for intersecting the long sets.
ii) Eclat algorithm does not make use of the Apriori property to reduce the candidate itemset generation
iii) The number of candidate itemsets generated in Eclat algorithm is much greater than that in Apriori algorithm.

For item sets with many dense and long patterns, there is huge performance overhead.

Eclat_Diffsets algorithm [13], adopts Boolean matrix implementation strategy to store the itemset and TID-Set .It then uses binary operation to calculate the intersection, which can obviously improve the efficiency of intersection.

In Diffsets approach only the differences in the TID_sets of a candidate itemset from its generating frequent itemsets is stored,which reduces memory requirements for storing intermediate results when the database is compact. HEclat algorithm [15],adopts a hash Boolean matrix implementation to store the TID_set of itemsets. It uses bitwise AND operation for calculation of intersection of itemsets. The method of hash Boolean matrix can only improve the efficiency only when the number of transactions of a database is not large. Eclat-opt algorithm [16], uses the technique of double layer hash table, partition list of the set of itemset and TID lost threshold techniques. These methods reduces the search space and improves the candidate generation time, and the time for intersection by clipping the candidate 3-itemset. The Eclat-opt algorithm is proved to be much more effective than other Eclat based algorithms. Some Algorithms assign weights to the transactional items. In weighted transaction database, a weight value is associated with each item in I, the set of items[17][18].

IV. RESULTS AND DISCUSSION

Table 1. *Comparison of FIM algorithms*

Sl No	**Comparion of Apriori,FP-Growth,Eclat**			
	Parameters	***Apriori***	***FP-Growth***	***Eclat***
1	Techniques used	Apriori Property,Join and Prune	FP-Tree Traversal	Set Intersection Method
2	Search Technique Used	BFS	DFS	DFS
3	Representation of Database	Horizontal	Hybrid	Vertical
4	Data Structures used	Array Based Candidate itemset	Tree Based FP Tree and Conditional FP tree	Array Based TID List
5	Increased Number of Items Per Transaction	Performance degrades over dense frequent item sets	Performs better than both algorithms at higher density	Performance better than apriori ,but degradation happens at higher Density
6	Increased No of transactions	Too much overhead	Almost the same Performance	Almost the same Performance
7	Candidate generation	Huge	No candidate itemsets are generated	Huge candidate generation
8	Memory Utilization	More Memory required due to huge candidate generation	Memory Utilization is low due to compact FP-Tree structure	Memory Utilization is low for small datasets
9	Number of Database Scans	Huge Number of Database scans	Two scan	Three scans
10	Databases	Suitable for sparse as well as dense datasets	Suitable for large and medium datasets	Suitable for medium and dense datasets,but not suitable for small datasets
11	Time Complexity	Huge for small data sets	Huge for Larger data sets	Huge for Larger data sets

V. CONCLUSION AND FUTURE SCOPE

Frequent itemset mining plays a very important role in association rule mining. It is a computationally intensive problem.In this survey paper ,a sincere effort is made to understand the various FIM algorithms,which serves as the core part of association rule Mining. A comparison study on the most widely used FIM algorithms such as Apriori, FP-Growth and Eclat conducted on various factors showed that FP-growth algorithm is suitable for medium-dense datasets and also the performance is better even with increased number of transactions .Also a study on the various strategies used for improvisation of these algorithms are done.

ACKNOWLEDGMENT

I would like to express my gratitude to my guide Dr.Prashant.CSR, Dean Academics ,New Horizon College of Engineering for his constant support, encouragement and valuable inputs.

References

[1] Rakesh Agrawal,Ramakrishnan Srikant,*"Fast algorithms for mining association rules"*, In the Proceedings of the 20th International Conference on Very Large Data Bases, VLDB, Santiago, Chile, pp 487-499, September 1994.

[2] Zaki, M. J. , *"Scalable algorithms for association mining". IEEE Transactions on Knowledge and Data Engineering*,Vol.12,Issue.3,pp 372–390,2000.

[3] Han,*"Mining Frequent Patterns Without Candidate Generation" in the* Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data. SIGMOD '00: pp 1–12,2000.

[4] Christian Borgelt,Xiaomeng Wang, *" SaM: A Split and Merge Algorithm for Fuzzy Frequent Item Set Mining",*

[5] J.Han,M.Kamber,"Data Mining Concepts and Techniques,Morgan Kaufmann Publisher,San Fransisco,CA,USA,2001.

[6] Zhang Changsheng, Li Zhongyue, Zheng Dongsong,*"An Improved Algorithm for Apriori"*,In IEEE,First International Workshop on Education Technology and Computer Science,2009.

[7] Gang Yang,Hong Zhao,Lei Wang,Yinng Liu *" Implementation of improved Apriori Algorithm"* in the proceedings of the Eighth International Conference on Machine Learning and Cybernetics, Baoding, 12-15 July 2009.

[8] Jianlong Gu, Baojin Wang , Fengyu Zhang, Weiming Wang, and Ming Gao *"An Improved Apriori Algorithm"* in the International Conference on Applied Informatics and Communication ICAIC 2011:Applied Informatics and Communication pp 127-133

[9] Cheung, D-W., Han, J., Ng, V-T., Wang, C-Y, *"Maintenance of Discovered Association Rules in Large Databases : An Incremental Update technique"* in the 12th International Conference on Data Engineering, New Orleans, LA., 26 February-1 March 1996,pp. 106-114.

[10] Quanzhu Yao, Xingxing Gao ,Xueli Lei and Tong Zhang , *"The Research and Improvement Based on FP-Growth Data Mining Algorithm"* in the Advances in computer Research,Vol.58 Modeling, Simulation and Optimization Technologies and Applications (MSOTA 2016) .

[11] Kuikui Jia,Haibin Liu, *"An Improved FP-Growth Algorithm Based on SOM Partition"* in the proceedings of International Conference of Pioneering Computer Scientists, Engineers and Educators ICPCSEE 2017: Data Science-pp 166-178.

[12] Ding Zhenguo, Wei Qinqin, Ding Xianhua *"An Improved FP-growth Algorithm Based on Compound Single Linked List"* in the 2009 Second International Conference on Information and Computing Science,IEEE ,DOI 10.1109/ICIC.2009.96

[13] M. J. Zaki and K. Gouda, *"Fast vertical mining using diffsets"*,in the Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining, New York, USA, (2003), pp. 326- 335.

[14] Caiyan Dai, Ling Chen, "An Algorithm for Mining Frequent Closed Itemsets with Density from Data Streams" ,International Journal of Computer Sciences and Engineering(IJCSE),Vol.4,Issue 2 ,pp.40-48,2016.

[15] X. Z. Yang, C. P. En and Z. Y. Fang, "*Improvement of Eclat algorithm for association rules based on hash Boolean matrix"*, Application Research of Computers, vol. 27, no. 4, (2010), pp. 1323-1325.

[16] F. P. En, L. Yu, Q. Q. Ying and L. L. Xing, "Strategies of efficiency improvement for Eclat algorithm", Journal of Zhejiang University (Engineering Science), vol. 47, no. 2, (2013), pp. 223-230.

[17] Akilandeswari. S, A.V.Senthil Kumar, "A Novel Low Utility Based Infrequent Weighted Itemset Mining Approach Using Frequent Pattern",International Journal of Computer Sciences and Engineering(IJCSE),Vol.3,Issue 7,pp.181-185,2015.

[18] R.B.M. Sayyad,P.S. Yalagi, "Infrequent Weighted Itemset Mining for Large Dataset" International Journal of Computer Sciences and Engineering(IJCSE),Vol.5,Issue 6,pp.149-153, 2017.

Authors Profile

Ms.Uma.N pursed Bachelor of Engineering from University College of Engineering, Thodupuzha, kerala in 2004 and Master of Technology from VTU in year 2011. She is currently pursuing Ph. D. and currently working as Assistant Professor in Department of Computer Science, New Horizon College of Engineering,Bangalore. Her main Research work focuses on Data Mining,Pattern mining etc.

Dr. Prashanth C.S.R pursed Bachelor of Engineering in computer science from Bangalore University,Master of Science from University of Texas at Dallas, USA and Ph.D from Auburn University,USA. Dr. Prashanth C.S.R is the Dean Academics, at New Horizon College of Engineering,Bangalore since May 2015.Prior, he was heading the Department of computer science,New Horizon college of Engineering since 2009.Prior to joining New Horizon college of Engineering, Dr. Prashanth C S R worked as an Assistant Professor in South University, USA for over 7 years. He has over 20 years of experience in both academics and in the Industry. Dr.Prashanth C.S.R has published extensively in international conferences and journals, and his major research interests include: Task Scheduling in Heterogeneous Computing Systems and High Performance Computing.