ISSN No. 0976-5697

Volume 2, No. 6, Nov-Dec 2011

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Graph Based Approaches to Generate Frequent Itemsets

P. Hari Shankar*
M. Tech, Scholar, Dept. of CSE
Aurora Technological and Research Institute
Hyderabad, India
harishankarpunna@gmail.com

S. Siva Sankar
Associate Professor, Dept. of CSE
Aurora Technological and Research Institute
Hyderabad, India
sivasankarssr@gmail.com

Abstract: Association Rule Mining among Frequent Items has been widely studied in Data Mining. Many researchers have improved the algorithm for generation of all the Frequent Itemsets. Frequent Itemset mining plays an essential role in Data Mining. Various algorithms have been proposed to generate all large frequent itemsets from a large amount of transaction data using graphs. In this paper we generally review and compare the most important graph based algorithms with each other. Results shows that each algorithm based on its applied strategy has some advantages and some disadvantages. However compress and mine algorithm is more effective and takes less time and space. In this paper we discuss various approaches to find Frequent Itemsets using Graphs.

Keywords: Data Mining; Frequent Itemset; Association Rule; Graph;

I. INTRODUCTION

Data Mining is to extract the unknown, potentially useful model or rule from large amount of data. Data Mining [1] started in late 80's, developed by leaps and bounds in 90's; it is still very active one in the forefront areas present.

Data Mining was also named as knowledge discovery from database. It can find potential, innovative and valuable information which can be understood by users.

Association Rule Mining, one of the most important and well researched techniques of Data Mining, was first introduced in [2]. It aims to extract interesting correlations, frequent patterns, associations or casual structures among sets of items in the transaction databases or other data repositories. Association rules are widely used in various areas such as telecommunication networks, market and risk management, inventory control etc.

Mining Association rules is for the discovery of meaningful correlation between sets or related links from large amount of data. Association Rule Mining is to find out association rules that satisfy the predefined minimum support and confidence from a given database. The problem is usually decomposed into two sub problems. One is to find those itemsets whose occurrences exceed a predefined threshold in the database; those itemsets are called Frequent or large itemsets. The second problem is to generate association rules from those large itemsets with the constraints of minimal confidence.

A database D consists of several records. Let L= {i_1, i_2 … i_n} denote a record of data items. An Itemset T is defined to be a subset of L. A k- itemset is a set with k items. The support or frequency of T is the percentage of records in D that contain the itemset. T is a frequent itemset if its support exceeds a given threshold. All subsets of a frequent itemset must also be frequent.

Given n items, there are potentially 2^n Itemsets, however, only a small fraction of them are usually frequent. The frequent itemsets discovery algorithms aim to find them quickly. In this paper, we focus on the various approaches to discover the frequent itemsets from the databases using Graphs.

Informally, a Graph is set of nodes, pairs of which might be connected by edges. In a wide array of disciplines, data can be intuitively cast into this format. A Graph can be applied to represent any physical situation involving discrete objects and a relationship among them. Graph representation can be used efficiently to associate items in a transaction to generate frequent itemsets in a very large database. In this paper we focus on various approaches to discover the frequent itemsets from the databases using graphs.

The Applications of Graphs [3] in Data Mining such as Chemical and Biological Applications, Web applications, Networking applications, Computer Network Applications. In Chemical and Biological Applications graphs are natural representations for their structures and used to find chemical compounds, used in structure driven prediction. Web applications include linkage structure of web, analysis of query flow logs. In Social networking applications we can perform community detections. In Computer network applications includes program control flow analysis, intrusion network analysis, mining communication networks.

This paper is organized as follows:

In Section-II we discussed preliminary concepts. In Section-III we discussed various graph based approaches for frequent itemsets generation. In Section-IV we compared performance of various graph based approaches. We concluded our final remarks in Section-V.

II. PRELIMINARY CONCEPTS

A. *Itemset:*

A set of items is referred to as an Itemset. An Itemset that contain K items is a K- Itemset.

B. *Frequent Itemsets:*

Let I= {i_1, i_2 …… i_n} be a set of literals called items. Let DB denote a set of transactions where each transaction T is a set of items, such that T is a subset of I. Associated with each transaction is a unique identifier, called Transaction Identifier

(TID). A set of items is referred to as an itemset. An itemset that contains K items is a K-itemset. The frequency of occurrence or support count of an itemset is the number of transactions that contain the itemset. An itemset satisfies minimum support if the frequency of occurrence of the itemset is greater than or equal to threshold. If an itemset satisfies minimum support then it is a frequent itemset.

C. Association Rule:

Association rules are statements of the form $\{x_1, x_2 \dots. x_n\} \Rightarrow Y$, meaning that if we find all of $x_1, x_2 \dots.. x_n$ in the market basket, then we have a good chance of finding Y. The probability of finding Y for us to accept this rule is called the *confidence* of the rule.

D. Graph:

A Graph G = (V, E) consists of a set of objects $V = \{v_1, v_2 \dots.\}$ called vertices and another set $E = \{e_1, e_2 \dots\dots\}$, whose elements are called edges such that each edge e_k is identified with an ordered pair (v_i, v_j) of vertices. In the graph based approach vertices represent the items and edges represent the association between various items to generate frequent itemsets.

III. RELATED WORK

A. Apriori Algorithm:

Apriori algorithm proposed by R. Agrawal and R. Srikanth in 1994 [4] for mining frequent itemsets for Boolean association rules. The problem of mining association between set of items in large databases was first introduced in the Apriori algorithm. Apriori employs an iterative approach known as a *level-wise* search, where k-itemsets are used to explore (k+1) – itemsets. It uses the generate-and-test approach to find all frequent patterns. Apriori algorithm is a famous algorithm for mining all frequent itemsets and uses Breadth-first search technique and a tree structure to count candidate itemsets efficiently. It generates candidate itemsets of length k from itemsets of length k+1. Then it prunes the candidates which have an infrequent sub pattern. The *Apriori Property* is any subset of frequent itemset must be frequent. According to the property of Apriori, many like Apriori algorithms had been proposed, but all algorithms had the bottle-neck that generating many candidates.

B. FP- Growth:

In order to solve the Apriori problem, J. Han and J. Pei proposed FP Growth algorithm [5] based on FP-tree that used the compressed FP-tree structure to store the frequent patterns and did not generate candidates. But in generation FP-Tree need to scan database twice. And the need of time also increased. FP Growth adopts a divide & conquers method to project databases based on the recently generated frequent patterns and grow longer patterns. Later Jun Gao proposed [6] a new association rule mining algorithm that is a modified FP Growth algorithm. It can convert a transaction database into a modified FP-tree through scanning the database only once, and then do the mining of the tree. All algorithms based on FP Growth process uses tree for arranging the items before mining, where more than one node can contain single item. This causes repetition of same item and needs more space to store many copies of same item.

In this paper we discuss different graph approaches which uses graph for arranging items before mining. The benefit of using graph is that there is only one node for an item. This requires less memory.

C. Graph Based Approaches:

a. *Show- Jane Yen and Arbee* [7] proposed a Graph based approach in 2001 for discovering various types of Association Rules from a large database of customer transactions. Their approach scans the database once to construct an association graph and then traverses the graph to generate all large itemsets. They proposed a uniform graph based approach to discover three types of Association Rules., that is, Primitive Association Rules, Generalized Association Rules, and Multiple-level Association Rules. A primitive Association Rule is an association rule which describes the association among database items which appear in the database. A generalized association rule which describe the association among items which can be generalized items or database items. The multiple-level association rules are discovered from a large database of customer transactions in which all items are described by a set of relevant attributes.

This approach includes the 5 phases.

a) *Numbering Phase:* In this all items are assigned an integer number.

b) *Large item generation phase:* This phase generates large items and records related information.

c) *Association Graph construction phase:* This phase constructs an association graph to indicate the association between large items.

d) *Association pattern generation phase:* This phase generates all association patterns by traversing the constructed association graph.

e) *Association Rule Generation phase:* In this phase the association rules are generated directly according to the association patterns.

Example1:

Table I: Transaction Database1

TID	Item set
100	3, 1, 4
200	5, 3, 2
300	1, 2, 3, 5
400	5, 2

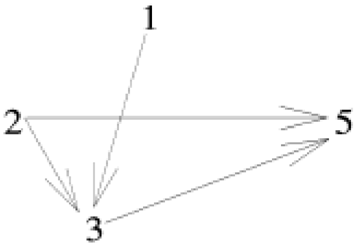


Figure. 1: Association Graph for Transaction Database 1

b. J. Chai, L. Jin, B. Hwang, etc.[8] all proposed a graph based approach in 2007 to mine the frequent patterns using Bipartite Grah. Bipartite Graph means a graph can be divided into two portions. In this approach they proposed a efficient ALIB algorithm (Algorithm using large items Bipartite Graph), that can find the frequent patterns by scaning the database only once. This can avoids the generation of candidate itemsets. This LIB-Graph compresses database information into a much smaller data structure. By this approach we can quickly find frequent patterns. This algorithm includes two steps. First, it seems the database once and founds the infrequent items, then LIB Graph structure is generated. This structure represents the relationship between items and transactions. Second, Finding frequent sets is done to verify if the number of transactions including the item sets is no less than a predifined minimum support threshold.

Example 2:

Table II: Transaction Database 2

TID	Items
100	f,a,c,d,g,I,m,p
200	a,b,c,f,l,m,o
300	b,f,h,j,o
400	b,c,k,s,p
500	a,f,c,e,l,p,m,n

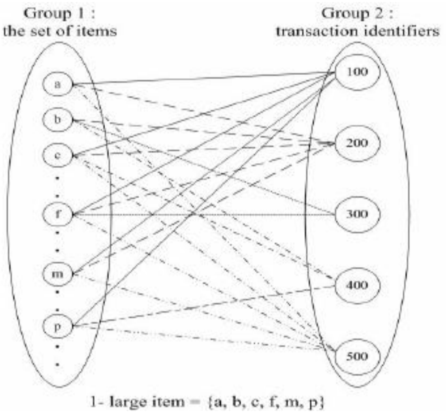


Figure. 2: LIB Graph for Transaction Database 2

c. *SHANG-ping Dai, DUAN Xin* [9] performed a research on Graph Based Algorithm. In their research they proposed two algorithms that is DLG (Direct Large Itemset Generation) and IODLG (Improved DLG Algorithm). They analyze the performance of both algorithms based on the minimum support value. The DLG algorithm proposed for generating large itemsets. There are 3 phases in DLG, Large 1-itemset generation, graph construction phase, large itemset generation phase. IODLG algorithm used databases have many short patterns, and this can obviously reduce the time accessing the adjacent nodes and producing frequent itemsets.

d. *Yufang Wang and Pei Tian* [10] proposed an improved grpah based algorithm in 2009 for mining association rules using complete Sub-graph. The improved algorithm based on fully sub-graph to find frequent itemsets, from which compared the improved algorithm with the original one. Basic idea of improved algorithm based on graph approach for mining association rules is, in a correlation graph of a database, any frequent k-item set always has a k-complete sub-graph corresponding. If there is a frequent k-item set, the k-items corresponding to the k nodes of the sub-picture derived a sub-graphs were found in the associated graph, all the frequent item sets were got.

When searching for k-complete sub-graph based on k-1 complete sub-graph, if degree of the incidence graph is less than k-1 node in the k-1 complete sub-graph, k-complete sub-graph based onk-1 sub-graph cannot be found; When k-1 complete sub-graph, in which degree of all nodes are more than adds a node, but the node does not belong to the complete sub-graph, and degree of it in the incidence graph is less thank-1, then k-complete sub-graph also cannot be able to get.

e. *Vivek Tiwari, Vipin Tiwari and Shailendra Gupta, etc.* [11] all proposed a graph based approach in 2010 for mining frequent Itemsets. They introduced an algorithm called FP-Growth-Graph which uses graph instead of tree to arrange the items for mining frequent itemsets. The benefit of using graph is that there be only one node for an item. This requires less memory. The proposed algorithm contains three main parts. The first is to scan the database only once for generating graph for all item. The second is to prune the non frequent items based on given minimum support threshold and readjust the frequency of edges, and then construct the FP-Graph. By this approach the space complexity is increased to get the frequent itemsets.

The FP-Graph consists of nodes and edges. Number of node in the graph is equal to number of distinct items in the database. Each node is associated with a value count which stores the number of occurrence of item in database. Each edge in graph contains three values marked on it. First value represents the frequency of edge, second value is the name of node with which concerned transaction has been started nad the third value represents the number nodes we have traversed to reach the destination of this edge including destination. By this process the graph can be constructed and then we prune the graph based the minimum support value and finally we can mine the frequent itemsets from the pruned graph.

Example 3:

Table III: Transaction Database 3

TID	List of Items
T001	A, B,E
T002	B, D
T003	B, C
T004	A, B, D
T005	A, C
T006	B, C
T007	A, B
T008	A, B, C, E
T009	A, B, C

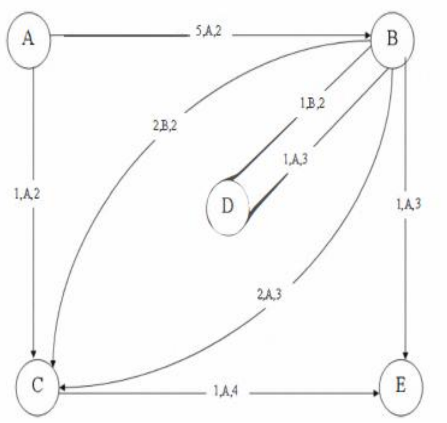


Figure. 3: Graph for Transaction Database 3

f. *P.Deepa Shenoy, Srinivas K.G, etc.* [12] all proposed an efficient Graph based algorithm to generate frequent itemsets using Compress and Mine technique. In this approach, once the data becomes huge, it increases number of I/O scans. Therefore, the usage of compression techniques on the databases minimizes the number of I/O scans and also reduces the space complexity. The objective of the proposed algorithm is to apply compression on the databases and identify frequent itemsets at single and multiple levels, in a transaction database using an efficient graph based approach. This algorithm scans the database once to find frequent one item and scans it again to construct an association graph to derive the path matrix, then traverses the path matrix to generate frequent itemsets. By using the Compress and Mine technique there is the improvement in both the time complexity and space complexity over the earlier graph based approaches.

The efficient graph based approach algorithm first divides the database into x segments, compress all segments and store in main memory. Then decompress the each segment one at a time. The decompressed segment is scanned to generate one-frequent items using minimum support. Then construct the association graph and path matrix. From the path matrix we can find the frequent itemsets. This approach is multiple level association rule mining.

Example 4:

Table IV: Transaction Database 4

TID	Items
1	a, c, e
2	b, c, e
3	a, b, c, e
4	b, e

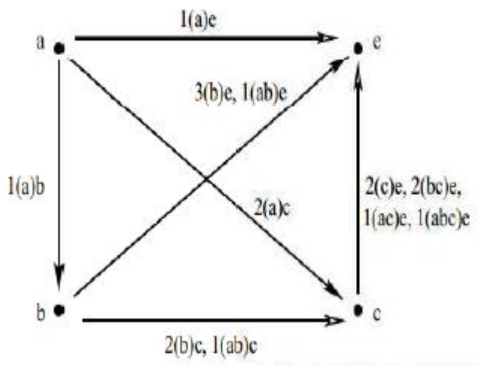


Figure. 4: Graph for the Transaction Database 4

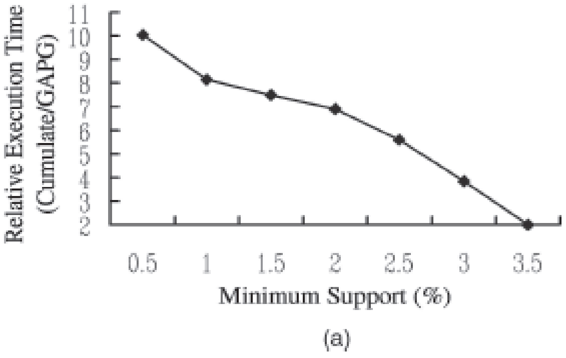
Table V: Path Matrix for Transaction Database 4

	a	b	c	e
a		1(a)b	2(a)c	1(a)e
b			2(b)c, 1(ab)c	3(b)e, 1(ab)e
c				2(c)e, 2(bc)e, 1(ac)e, 1(abc)e
e				

IV. COMPARISON

Here we compare the different algorithms which are already proposed and compare the algorithms in various aspects.

From [7] the Relative execution time is compared with respect to the minimum support values and the number of transactions. By this comparison we can evaluate the performance of the proposed algorithm which is showed in Fig. 5.



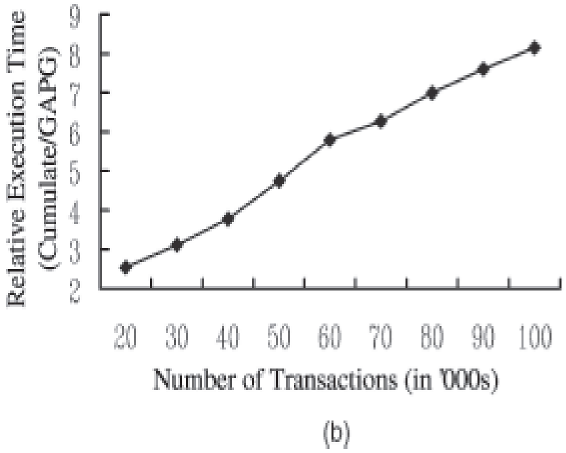


Figure. 5: Relative Execution Time (Reference [7]).

From [8] the performance of the both algorithms that is ALIB and FP- Growth is represented in Fig. 6 the runtime of ALIB and FP- Growth on synthetic datasets as the support threshold decreases from 0.3% to 0.01%. As shown in figure both ALIB and FP- Growth achieve good performance in the frequent pattern mining. ALIB is faster than FP- Growth.

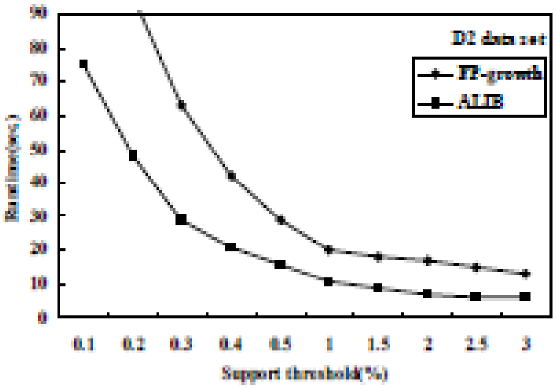


Figure. 6: the performance comparison of ALIB and FP-Growth

From [9] the performance of the both IODLG and DLG approaches is represented in Fig. 7. The IODLG algorithm in the databases reduces the time accessing the adjacent nodes and producing frequent itemsets. Especially when the length of frequent itemsets and the output value of nodes were increased, the algorithm will be more effective and feasible.

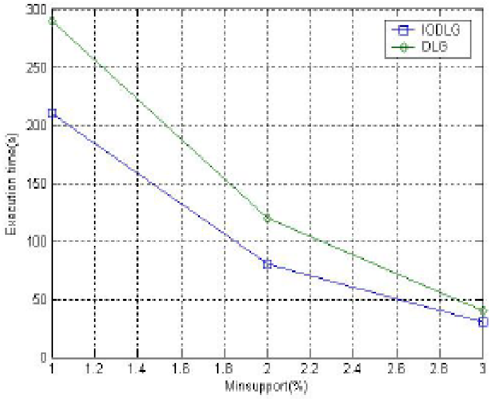


Figure. 7: performance comparison of IODLG and DLG

From [11] we can compare the total processing time for a database with both approaches that is FP- Tree and FP- Graph is showed in Fig. 8.

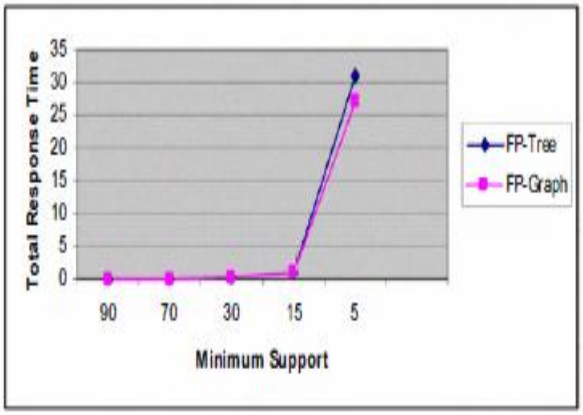


Figure. 8: Total processing time for a Database

V. CONCLUSION

In this paper we discussed various approaches for Frequent Itemsets generation using Graphs. Frequent Itemset generation is a data mining technique that involves finding frequent items. We discussed various algorithms which uses binary transactional databases. We compared the performance of various graph based approaches for frequent itemset generation. In our future work we will concentrate on graph based approach using Non- Binary Databases.

VI. REFERENCES

[1] Han Jiawei, Data Mining: Concepts and Techniques, Burnaby: Simon Fraser University, 2000., pp. 155-163

[2] Agrawal R.; T. Imielinski; A. Swami: " Mining Association Rules Between Sets of Items in Large Databases", SIGMOD Conference 1993: 207-216.

[3] Charu C. Aggarwal, H. Wang: " Managing and Mining Graph data", Kluwer Academic Publishers, London, 2010, p.p. 43-51.

[4] Agrawal R., and R. Srikanth, " Fast algorithms for mining association rules", In VLDB'94, pp.487-499, 1994.

[5] Han J., " Mining frequent patterns without candidate generation", Data Mining and Knowledge Discovery, Kluwer Academic Publishers, pp. 53-87, 2004.

[6] Gao Jun: "A New Algorithm of Association Rule Mining" 2008 International Conference on Computational Intelligence and Security, pp. 117-121.

[7] Show-Jane Yen and Arbee L.P. Chen, Member, IEEE: " A Graph-Based Approach for Discovering Various Types of Association Rules" 2001 IEEE Transactions on Knowledge and Data Engineering Vol 13, No. 5.

[8] Duck Jin Chai, Long Jin; Buhyun Hwang; Keun Ho Ryu: "Frequent Pattern Mining using Bipartite Graph", 2007, 18th International Workshop on Database and Expert Systems Applications, pp-182-186.

[9] Shang-ping Dai, Duan Xin: " Reasearch on Graph- Based Algorithm", 2008, International Symposium on Computational Intelligence and Design, pp:17-20.

[10] Yufang Wang, Pei Tian: “An Improved Algorithm for Mining Association Rules Based Complete Sub-graph”, 2009 IEEE, pp. 1-4.

[11] Vivek Tiwari, Vipin Tiwari, Shailendra Gupta, Renu Tiwari: “ Association Rule Mining: A Graph Based Approach for Mining Frequent Itemsets”, 2010 International Conference on Networking and Information Technology, pp: 309-313.

[12] P. Deepa Shenoy, Srinivasa K. G, Achint O Thomas Venugopal K. R, L. M. Patnaik: “ Compress and Mine: An Efficient Graph Based Algorithm to Generate Frequent Itemsets”,2004, pp. 1-10.