GRAPH BASED NEW APPROACH FOR FREQUENT PATTERN MINING

Anurag Choubey[1] , Dr. Ravindra Patel[2] and Dr. J.L. Rana[3]

[1]Dean Academic, Technocrats Institute of Technology, Bhopal, (M.P.), India
E-mail: anuragphd11@gmail.com
[2]Reader & Head, Department of Computer Application, UIT-RGPV, Bhopal, (M.P.), India
E-mail: ravindra@rgtu.net
[3]Former Professor & Head, Department of Computer Science & Engineering, MANIT, Bhopal, (M.P.), India
E-mail: jl_rana@yahoo.com

ABSTRACT

Association rule mining is a function of data mining research domain and frequent pattern mining is an essential part of it. Most of the previous studies on mining frequent patterns based on an Apriori approach, which required more number of database scans and operations for counting pattern supports in the database. Since the size of each set of transaction may be massive that it makes difficult to perform traditional data mining tasks. This research intends to propose a graph structure that captures only those itemsets that needs to define a sufficiently immense dataset into a submatrix representing important weights and does not give any chance to outliers. We have devised a strategy that covers significant facts of data by drilling down the large data into a succinct form of an Adjacency Matrix at different stages of mining process. The graph structure is so designed that it can be easily maintained and the trade off in compressing the large data values is reduced. Experimental results show the effectiveness of our graph based approach.

KEYWORDS

Data mining, Frequent pattern, Graph structure, Adjacency Matrix

1. INTRODUCTION

Data mining is the non-trivial mining of implicit and structured data, previously unknown prototype that is potentially useful especially decision making systems. It is one such area which continues to pave its way from bioinformatics [1] to web-based mining[2], from image appreciation[3] to wireless remote sensing[4]. Be it any giant organization or any small private run firm, they all need to figure out associative relationships for improving marketing strategies and informed business decisions.

Along with major domains like sequential patterns, classifications, clustering etc, Association rule discovery has been an active area of examination. Frequent pattern mining was first introduced by Agrawal et al. [5] The methods for efficient exploration is based on the candidate generation-and-test approach like Apriori [6,7] and pattern growth strategy during the series of decades [8,9,10,11,12] have already been developed but they have a huge setback that generates n number of candidate sets and takes as many database scans in large datasets for frequent itemsets discovery. Also the performance gets degraded when the size of database is exponential. To improve efficiency of mining process, Han et al [13,14,15] proposed an alternative frame work, namely a tree based frame work. The algorithm F. P. growth, they

DOI : 10.5121/ijcsit.2012.4117

proposed in this frame work construct an extended prefix tree structure, called frequent pattern tree (FP tree) to capture the content of the transaction data base. Thus requires only two full I/o scans of the dataset to build the prefix tree in main memory and then mines pattern directly from structure. Performance studies demonstrate that the method substantially reduces search time but massive creation of conditional trees makes this algorithm not scalable to mine large data sets beyond few millions.

Our new approach will not only enhance the performance but also absorb significant amount of time as well as money spent on serial scans thereby diminishing the bottleneck arising in variable length database containing huge number of candidate sets and long patterns. Through this paper, we are interested in finding a graph-based approach so as to find the frequent pattern that repeatedly appears across various transactions. The data-mining model that we are using here is adjacency matrix and a graph since a graph is the simplest mean to represent and discover informative frequently occurred set of items.

2. RELATED WORK

Graphs become increasingly useful in modeling complicated structures like circuits, images, protein structures, biological networks, chemical compounds. Graph mining has gain importance in data mining. Many researchers have developed various algorithms on graph mining. Some of them are discussed in this section. In the beginning, in 1976 Ullmann [16] introduced an algorithm for subgraph isomorphism. This algorithm attains efficiency by inferentially eliminating successor's nodes in the tree search. In 1994 cook and holder [17,18] discovering a new version of their SUBDUE substructure system based on minimum length principal. Yen and Chen [19] in 1996, proposed graph based algorithm DLG, to efficiently solve the problem of mining association rule. Inokuchi, Washio and Motoda [20] in 1998 proposed a novel approach name AGM to efficiently mine the association rule among the frequently appearing substructure in a given graph dataset. Huan, Wang and Prince [21] in 2003 proposed a novel subgraph mining algorithm FFSM, which employs a vertical search scheme. Since then many graph mining algorithms [22,23,24] have been developed with improvement in efficiency. Recently in 2010, [25] Jiang et al suggested three strategies for controlling candidate set generation. In 2010 Pradeep Chouksey, R.S. Thakur & R.C. Jain [26] presented a graph based approach for exploring multilevel frequent pattern from transactional databases. At each level it scans the datasets once and creates a directed graph, which is stored in from of an adjacency matrix and calculates all frequent patterns at the same level. In 2011 [34], we presented a novel approach labeled CEG & REP by constructing graph structure used for mining only closed sequences.

3. PROBLEM SPECIFICATION

Association rule mining as introduced in [5] searches for relationship between items in a data set. It finds association, correlation, or casual structures among set of items or objects in transaction databases, relational databases and other information repositories. To mine an association rule, database of transaction is needed. And each transaction is list of items. Then apply mining algorithm to find the association rule.

Finding frequent itemsets plays an impotent role in the field of data mining. Frequent item set are essential for many data mining problems like discovery of association rule [27,28] correlation [29,30] and sequential pattern [31,32].

As defined in [7], the problem is stated as follows.

Let I = {x_1,….xn} be a set of items. An itemset X is a subset of items, i.e. X $\subseteq$ I. A transaction T = (itd, X) is a 2-tuple, where tid is a transaction-id and X an itemset. A transaction T=(tid, X) is said to be contain itemset Y if and only if Y is a subset of X. A transaction database D is a set of transactions. The number of transactions in D containing itemset X is called the support of X. Given a transaction database D and a support threshold *min_sup*, an itemset X will be called as frequent pattern if and only if sup(X)≥*min_sup*.

4. PRE-REQUISITES IN GRAPH BASED APPROACH

For better understanding, there are some mathematical terms that need to be discussed before proceeding on to our research as given in [33]. The below mentioned terminology is frequently used in this paper.

Graph

A linear graph G = (V,E) consists of a set of objects V = { v_1,v_2 } called vertices and another set E = { e_1, e_2 } whose elements are called edges, such that each edge is e_k identified with an unordered pair (v_i , v_j) of vertices . The vertices v_i , v_j associated with edge e_k are called the end vertices of e_k.

Directed Graph

In a graph G = (V , E), an edge which is associated with an ordered pair of V * V is called a directed edge of G. A graph in which every edge is directed is called a directed graph.

Adjacency matrix

The adjacency matrix of a graph G with n vertices and no parallel edges is an n by n symmetric binary matrix X = [x_{ij}] defined over the ring of integers such that

x_{ij} = 1 , if there is an edge between i_{th} and j_{th} vertices

= 0 , if there is no edge between them.

In this paper, we made modification in the definition of adjacency matrix for a graph with parallel edges, is a n by n symmetric matrix X = [x_{ij}] defined as

x_{ij} = m, and m >0 if there are m directed edges between vertices i to j.

= 0, if there is no edge between them

5. PROPOSED GRAPH BASED APPROACH

One of the best studied data structure in computer science and discrete mathematics are graphs. In section 2 we have already discussed some of the related work already done making use of graph structure. It can therefore be stated that graph based data mining has became quite popular in the last few years [26,34,35,36]. Time complexity is one of the major issue in frequent itemset mining and our proposed new approach is towards the solution of it. Surprisingly, our graph based approach scan the whole database just once, and this salient feature is the most sought after paradigm in the domain of identifying frequent sets that results in generating large number of candidate sets. Preceded by the scanning, it constructs a graph, which is basically a directed graph. The weights of the graph are stored in the main memory and are represented in the form of an adjacency matrix. Here the items from the dataset are shown through the vertex of the directed graph and the weight of the vertex represents the support count of one itemset rather than multiple set. The vertices containing weights are connected by an edge. Relationships must be defined in case of mining large k-itemsets (k>=3).

It may happen that many transactions in a database might contain the same set of items, even if two transactions are radically different from each other and the case may be that two transactions may contain identical itemsets, thus their subsets may be common.

5.1 Working of the graph based approach

To evaluate frequently mined items, we consider these two semantics:

1. Fully qualified adjacency matrix
2. Reduced matrix close to frequent itemsets

To understand how this method is going to work, we use the below mentioned Transactional dataset. The dataset (Fig. 1) contains 6 transactions with namely, Transactional ID's, Items and their occurrences.

TID	a	b	c	d	e
1	0	0	0	0	1
2	0	6	0	1	1
3	2	0	2	0	1
4	0	0	0	0	0
5	1	0	6	0	1
6	1	1	1	0	0

Figure 1: Data Table of transactional database D

5.1.1 Fully qualified adjacency matrix

Firstly, it is going to scan the database D and will construct a directed graph G that is shown in fig. 2.

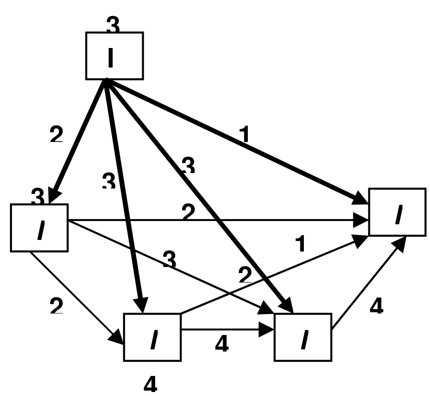

Figure 2: Directed Graph *G* of Database *D*

The items are kept in the database in lexicographical order and an edge having direction is made in special form to connect the two vertices.

For instance, if there is an edge I2→I4 i.e. as according to our transactions, occurrence of I2 is always preceed I4 and if I2 and I4 are numeric number, then I2<I4. The thing to be

noted here is that the edge (I2, I4) and (I4, I2) is considered to be identical. Therefore we obtained a symmetric matrix.

The directed graph G (Fig. 2) has been stored in memory by making it into the form of an adjacency matrix (Fig. 3) as shown below.

$$A = \begin{array}{c} \\ I1 \\ I2 \\ I3 \\ I4 \\ I5 \\ I6 \end{array} \begin{array}{c} \begin{array}{cccccc} I1 & I2 & I3 & I4 & I5 & I6 \end{array} \\ \left(\begin{array}{cccccc} A_{11} & A_{12} & A_{13} & A_{14} & A_{15} & --- \\ & A_{22} & A_{23} & A_{24} & A_{25} & --- \\ & & A_{33} & A_{34} & A_{35} & --- \\ & & & A_{44} & A_{45} & A_{46} \\ & & & & A_{55} & A_{56} \\ & & & & & A_{66} \end{array} \right) \end{array}$$

Figure 3: Adjacency Matrix of Directed Graph G

5.1.2 Reduced matrix close to frequent itemsets

Secondly, it verify the value count of each element of the matrix A_{ij}, and for any diagonal element (for i=j) A_{ij} < min_sup, the row and column of corresponding element is deleted from the matrix since the fact is that any super set of any infrequent itemset will never be frequent. Thus, a zero is assigned i.e. for $A_{ij.count} = 0$. After performing the second step, we get a filtered adjacency matrix as given in fig. (4). Here adjacency matrix can be lower or upper triangular matrix since the matrix is symmetric matrix. We are considering upper triangular matrix.

$$A = \begin{array}{c} \\ I1 \\ I2 \\ I3 \\ I4 \\ I5 \end{array} \begin{array}{c} \begin{array}{ccccc} I1 & I2 & I3 & I4 & \end{array} \\ \left(\begin{array}{ccccc} A_{11} & A_{12} & A_{13} & A_{14} & A_{15} \\ & A_{22} & A_{23} & A_{24} & A_{25} \\ & & A_{33} & A_{34} & A_{35} \\ & & & A_{44} & A_{45} \\ & & & & A_{55} \end{array} \right) \end{array}$$

Figure 4: Filtered Adjacency Matrix

Following is the frequent itemsets calculation list

A_{12} .list ∩ A_{13}.list = {*I*1, *I*2, *I*3} = (T1, T2)

A_{12} .list ∩ A_{14}.list = {*I*1, *I*2, *I*4} = (T1, T2)

A_{13} .list ∩ A_{14}.list = {*I*1, *I*3, *I*4} = (T1, T2, T3)

Similar from second row, calculate 3-itemsets

A_{23} .list ∩ A_{24}.list = {*I*2, *I*3, *I*4} = (T1, T2)

A_{23} .list ∩ A_{25}.list = {*I*2, *I*3, *I*5} = (T1)=1, it is less than min_sup thus, it is not frequent

A_{24} .list ∩ A_{25}.list = {*I*2, *I*4, *I*5} = (T1, T4)

From this row, calculate 3-itemsets

A_{34} .list ∩ A_{35}.list = {*I*3, *I*4, *I*5} = (T1, T5)

For frequent 4-itemsets calculate from first row

A_{12} .list ∩ A_{13}.list ∩ A_{14}.list = {*I*1, *I*2, *I*3, *I*4} = (T1, T2)

Similarly calculate from second row
A_{23} .list ∩ A_{24}.list ∩ A_{25}.list = {I2, I3, I4} = (T1, T2),
It is less then min_sup thus, it is not frequent

Ultimately, we find all the frequent patterns generated after sorting out the reduced adjacency matrix. We obtain frequent-1 itemsets, frequent-2 itemsets and frequent-3 itemsets as given in fig. (5).

TID	a	b	c	d
1	0	0	0	0
2	0	6	0	1
3	2	0	2	0
4	0	0	0	0
5	1	0	6	0

Figure 5: Reduce Data Table

Having performed these steps, we finally get the following generated frequent itemsets at concept

level-1 minsup=4

Large-1 itemset = (a1}:1
= {c2}:1

Large-2 itemset ={c2,d1}:2
= {c2, a1}:2

Similarly, for generating more frequent lower level patterns, the procedure is repeated as at goes along with each level, that means at each conceptual pattern it will again scans the datasets one more time and creates a directed graph in the form of adjacency matrix, thereby generating all frequent patterns which are filtered. The figure 6 is made by reducing those sets of items which are proved to be least occurring and thus called as infrequent set

Level-2 minsup = 3

Large-1 itemset = {a1}:1
= {b6}:1
= {c2}:1
Large-2 itemset = {a1,b6}:2
= {c2, d1}:2
= {e1, d1}:2
= {c6, d1}:2

Large-3 itemset = {a1,a2,b1}:3
= {c6,a2,b6}:3
= {c1,d1,b1}:3

Figure 6: Reduced Data set

Now finally all the patterns generated with the next level are generated and are shown below:

Level-3 minsup = 3

Large-1 itemset = {a1, b6,a2}:1
= {a1,c2,d1}:1
= {a1,d1,c6}:1
Large-2 itemset = {a1,b6,c2,d1}:2

5.2 Algorithm for Graph based approach

Input: The set of transaction D and total number of itemsets and their occurrences.

Output: Frequent itemsets

Our algorithm works in these many phases:

a) Scan: scan the database D and initialize it in the form of adjacency matrix

b) Identify: identify and update the values of each element of matrix A_{ij} and A_{ij}.count

c) Construct: construct the reduced adjacency matrix by removing corresponding row and column of element A_{ij}.count = 0 for i=j only. Frequent-1 itemset is generated indicating the frequently mined patterns.

d) Mine: more level will me mined at this step from each row by using logical AND operator between row elements.

6. EXPERIMENTAL RESULTS AND COMPARISON

To study the performance of the Graph based approach, we have implemented this algorithm in C++ and tested on a PIV machine with 128 MB RAM and Windows/XP.

To find experimental results we used Retail Store Dataset obtained from http://fimi.cs.helsinki.fi/data/, and another synthetic data set which is created by us.

6.1 Experimental results with retail store data set:

We used retail store data set obtained from *http://fimi.cs.helsinki.fi/data/,* the data generator is available from *http://www.almaden.ibm.com/cs/disciplines/iis/*. This dataset contains the retail data with 5000 transactions and 76 items. This dataset is obtained from Belgian Retail Store and is available for research purpose. With this data set we compared our graph based approach with apriori, partition & FP-Tree algorithm.

fig 7 shows the experimented count ratio performance of our graph based approach with these algorithms. Clearly our graph based approach shows the effectiveness with minimum count ratio as compare to apriori, partition & FP-Tree algorithm, as it captures the whole data set in only one scan.

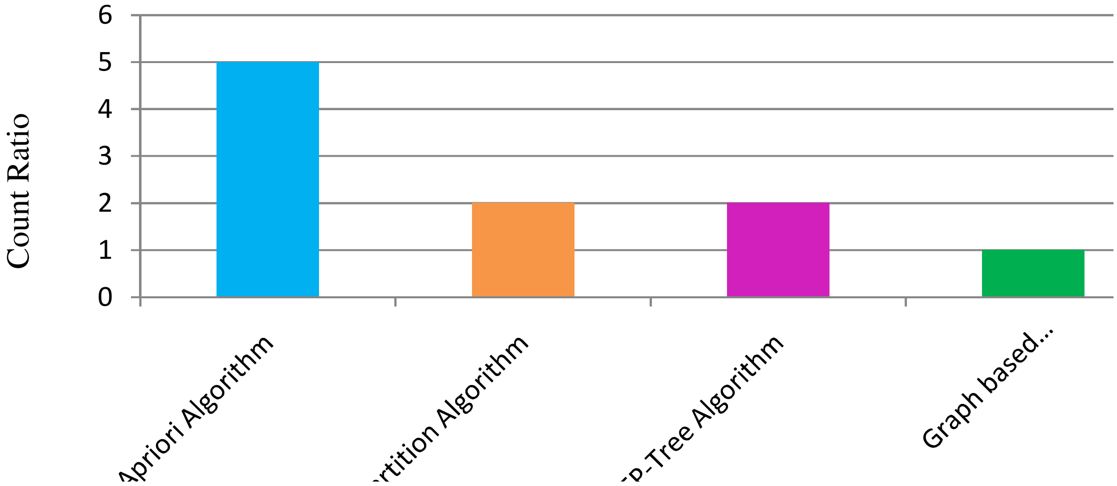

Figure 7: Count Ratio Performance Comparison

fig. 8 shows the execution times with partition, graph based approach & FP-tree algorithms. The minimum support is varied between 5% and 40%. The result shows that graph based approach is faster than partition algorithm and it is comparable with FP- tree algorithm. The performance gain is achieved by the significant reduction of the number of candidates.

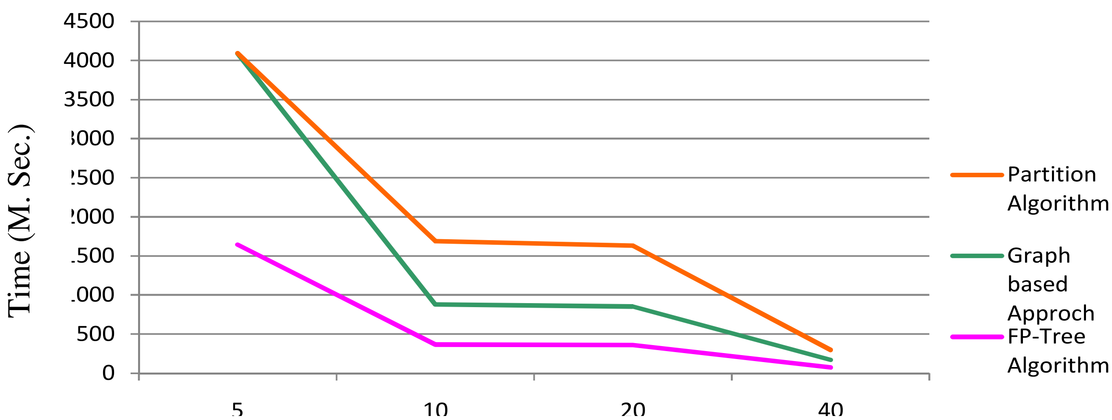

Figure 8: Comparison of Graph Based Approach with partition & FP tree algorithm

6.2 Experimental results with another synthetic data set:

We obtained experimental results with another synthetic data set, which is created by us. This dataset is very large and dense database. The parameter setting of this dataset is given in fig. 9.

Name	|*T*|	|*I*|	|*D*|	Size
Own Data set	5	3	100,000	2.11 MB

Figure 9: Parameter Setting

6.2.1 Experiment and comparison with Apriori algorithm

Fig.10 shows the running time of Graph based multilevel frequent pattern mining approach with Apriori based multilevel frequent pattern mining algorithm on our created database with respect to the minimum support threshold at level 1. The minimum support at level 2 and level 3 are fixed to 5% and 3% respectively

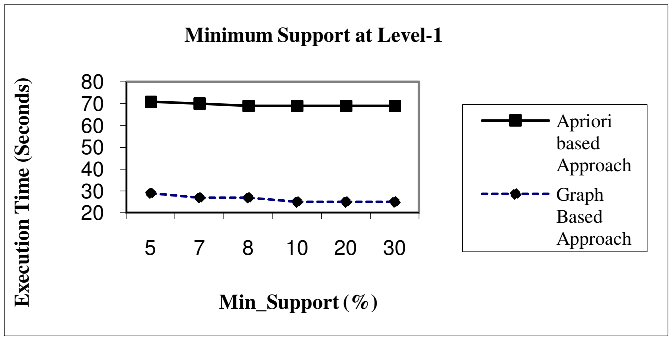

Figure 10: Threshold at level-1

Fig.11 shows the running time of these algorithms with respect to the minimum support threshold at level 2. The minimum support at level 1 and level 3 are fixed to 30% and 1% respectively. Similarly Fig.12 and 13 shows the running time of these algorithms with respect to the minimum support threshold at level 3 and 4. The minimum support at level 1 and level 2 are fixed to 30% and 3% respectively.

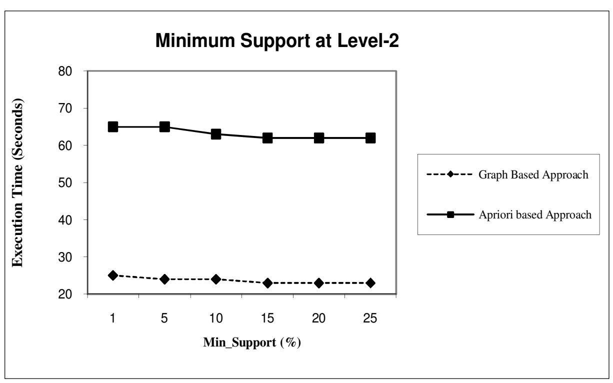

Figure 11: Threshold at level-2

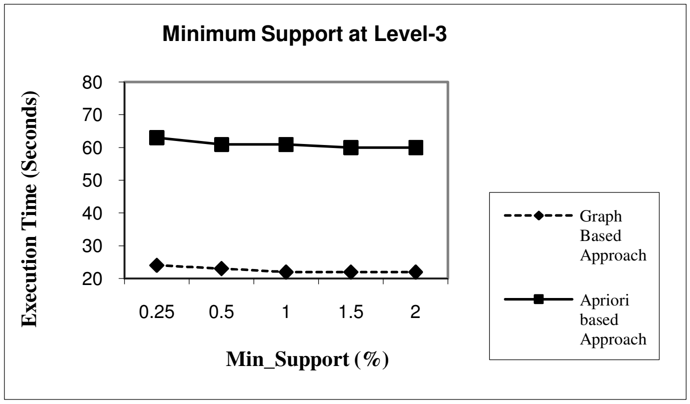

Figure 12: Threshold at level-3

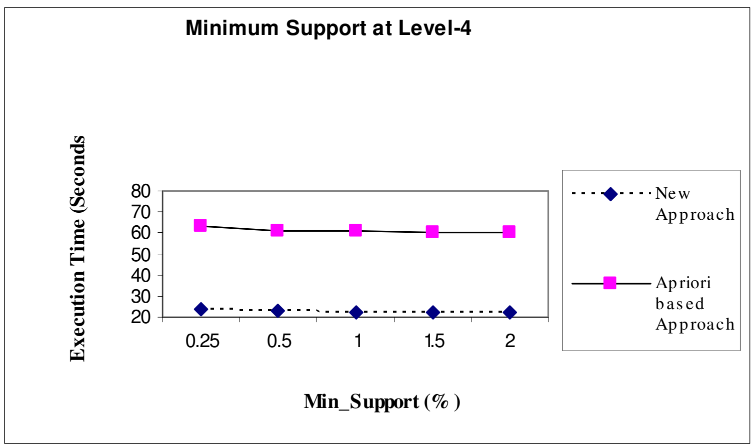

Figure 13: Threshold at level-4

We have conducted level wise experiment with our own data set for apriori and our new graph based approach and observed the execution time with respect to different minimum support. Experimental results show that our new approach is faster than apriori and maintaining the wide gap of execution time at each level.

6.2.2 Experiment and comparison with FP-growth algorithm

We also have conducted experiment with FP-growth algorithm.

Fig.14 shows the running time of Graph based multilevel frequent pattern mining approach with FP-growth algorithm on our created database with respect to the minimum support threshold at level 1. The minimum support at level 2 and level 3 are fixed to 5% and 3% respectively

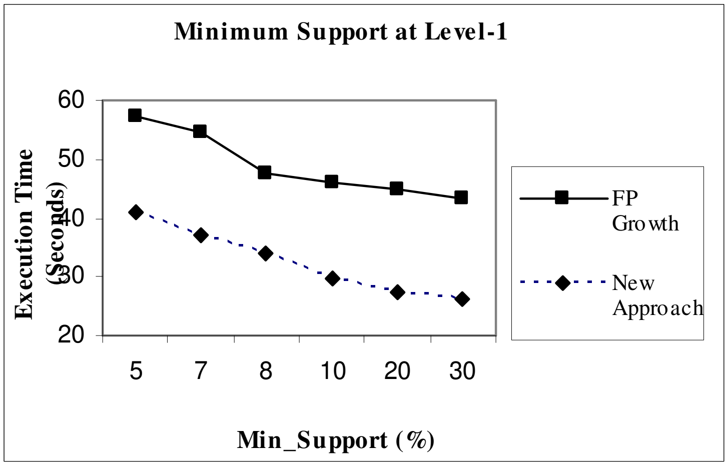

Figure14: Threshold at level-1

Fig.15 shows the running time of these algorithms with respect to the minimum support threshold at level 2. The minimum support at level 1 and level 3 are fixed to 30% and 1% respectively. Similarly Fig.16 and 17 show the running time of these algorithms with respect to the minimum support threshold at level 3 and 4. The minimum support at level 1 and level 2 are fixed to 30% and 3% respectively.

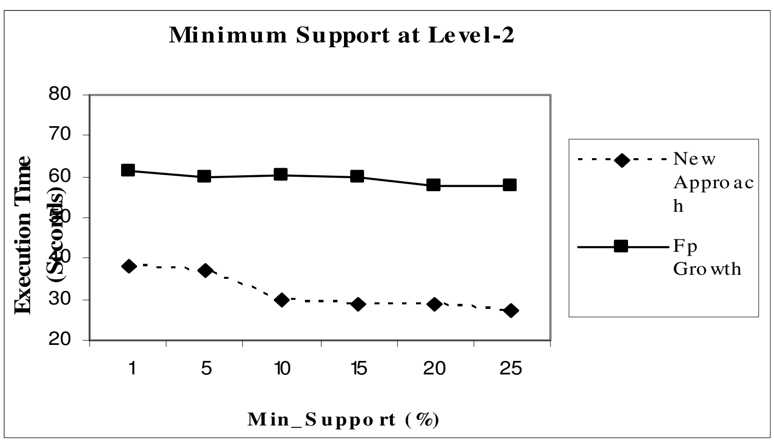

Figure15: Threshold at level-2

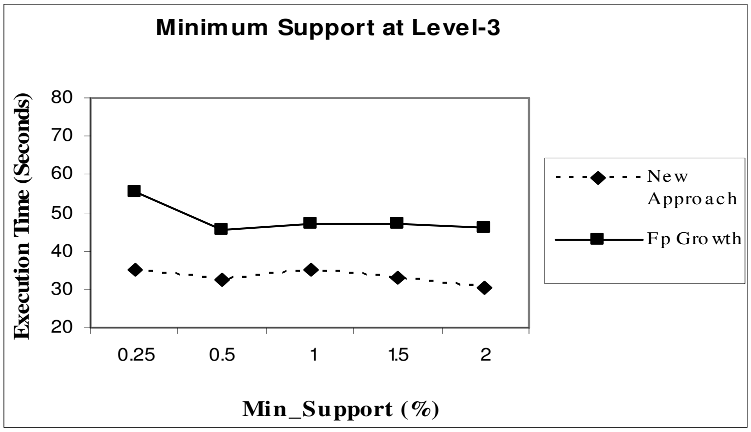

Figure 16: Threshold at level-3

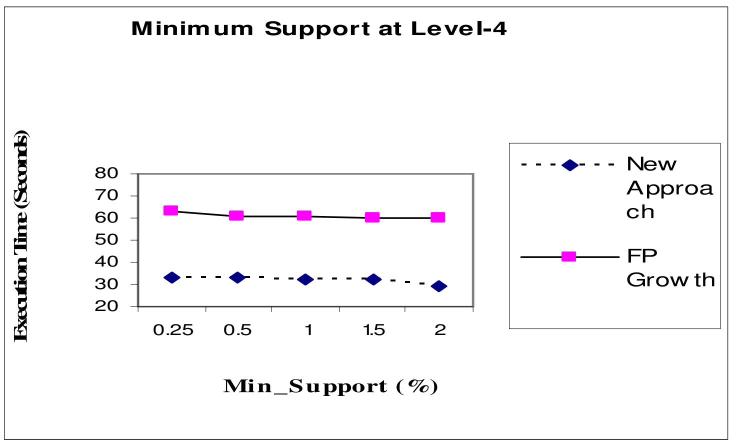

Figure 17: Threshold at level-4

The result shows as minimum support decreases, the running time increases. The new approach running faster than the existing algorithms Apriori and FP growth because, it required only one database scan at each level. As our database is dense and sparse, a large number of frequent patterns can be mined.

7. Conclusion

In this paper, we introduced a novel, fast approach for estimating the procedure of identification of frequent itemsets, which makes full use of graph theory. Graph is an efficient way to represent and understand the complex data, which has the potential to depict the values in a precise diagrammatic manner for dense and sparse databases. Experimental results show the better performance of our new approach as compare to not only with apriori but also proven to be faster than partition algorithm and maintaining better and comparable performance with FP-growth algorithm. One interesting thing to know about adjacency matrix here in our research is that though it occupies larger area of memory at initial stages, it releases those occupied area as soon as we delete the infrequent itemsets. Thus reducing I/O overhead and avoids the series of recursive scans. It is an improvement over the traditional frequent pattern mining algorithms but it has got enough room for further modifications and refinements.

REFERENCES

[1] H. Chen, (2005) *Intelligence and Security Informatics for National Security: Information Sharing and Data Mining*, to be published, Springer.

[2] J. Srivastava et al., (2000) Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data," *SIGKDD Explorations*, vol. 1, no. 2, pp. 12-23.

[3] M. Xu, L. Wu, (2004) Research on Remote Sensing Image Data Mining Prototype System and the RSIDMM-DTM, 0-7803-8742-2/04/$20.00 (C) IEEE

[4] Zhao P., Di L. (2006) Semantic Web Service Based Geospatial Knowledge Discovery. *Proceedings of 2006 IEEE International Geoscience* and Remote Sensing Symposium. Denver, Colorado, USA.

[5] R. Agrawal, T. Imielinski nd A. Swmi, (1993) Mining association rules between sets of items in large databases, in Proc. CM-SIGMOD Int. conf. pp. 207-216.

[6] R. Agrawal, H. Mannila, R. Srikant, H. Toivonen and A Verkamo, (1996) Fast Discovery of Association Rules, in Advance in Knowledge Discovery and Data Mining, MIT, pp. 307-328.

[7] Agrawal R. and Srikant R. (1994) Fast algorithms for mining association rules in large databases. In *Proc. 20th VLDB*, pages 478-499.

[8] R.J. Bayardo, R. Agrawal and D. Gunopulos, (1999) Constraint-Based Rule Mining in Large, Dense Databased," Proc. Of the 15th Int'l Conf. on Data Engineering (ICDE).

[9] H. Mannila, H. Toivonen, and A. I. Verkamo, (1994) Efficient Algorithms for Discovering Association Rules, in KDD94, pp. 181-192.

[10] J. Pei, J. Han, H. Lu, S. Nishio,s Tang, and D. Yang, (2001) H.Mine: Hyper-structure Mining of Frequent Patterns in Large Databased, in Proc. The JEEE Int'l Conference on Data Mining, pp. 441-448.

[11] J. Peo, J. Han and Lakshmanan, (2001) Mining Frequent Itemsets with Convertible Constrants, in ICDE.

[12] A. Sarasere, E. Omiecinsky, and S. Navathe, (1995) An efficient algorithm for mining association rules in large databases," in Proc. 21st VLDB, pp 432-444.

[13] J.Han, J. Pei, Y. Yin. (2000) Mining frequent patterns without candidate generation, In Proc. SIGMOD, pp. 1-12

[14] J.Han, J. Pei, Y. Yin, and R. Mao. (2004) Mining frequent patterns without candidate generation: a frequent-pattern tree approach, Data mining and Knowledge Discovery, 8(1), pp. 53-87.

[15] Han. J. etal, (2007) Frequent pattern mining; current status and future directions, Data Mining and Knowledge Discovery, vol. 15[1], pp 55-86.

[16] J. R. Ullmann, (1976) An algorithm for subgraph isomorphism. J. ACM, 23, pp. 31-42.

[17] D. J. Cook and L. B. Holder, (1994) Substructure discovery using minimum description length and background knowledge, Journal of Artificial intelligence Research, 1, 231-255.

[18] Holder, L. B. Holder, Cook, D. J. Cook, Djoko, S. Djoko, (1994) Substructure Discovery in the SUBDUE system, In Proc. AAAI'94 Workshop knowledge Discovery in Databases (KDD'94), pp 169-180.

[19] S.J. Yen and A.L.P. Chen. (1996) An Efficient Approach to Discovering Knowledge from Large Databases. In Proc. Of the IEEE/ACM International Conference on Parallel and Distributed Information Systems, Pages 8-18.

[20] A. Inokuchi, T. Washio, H. Motoda, (1998) An Apriori-based Algorithm for Mining Frequent substructures from Graph Data. In proc. 2000 European Symp. Principle of Data mining and knowledge Discovery (PKDD'00), pp. 13-23.

[21] J. Huan, W. Wang and J. Prins, (2003) Efficient Mining of frequent Subgraph in the Presence of Isomorphism. In Proc. 2003 int. conf. Data mining (ICDM'03), pp. 549-552.

[22] H. Hu, X. Yan, Y. Huang, J. Han and X. J. Zhou, (2005) Mining coherent dense subgraphs across massive biological networks for functional discovery. In proc. 2005 Int. conf. Intelligent system for molecular Biology (ISMB'05), pp. 213-221.

[23] A.M. Fahim, G. Saake, A.M. Salem, F. A. Torkey, M.A. Ramadan, (2008) K-mens for spherical clusters with large variance in sizes. WorldAcademy of science, Engineering & Tech., 45, pp. 177-182.

[24] S. Maji, S. Mehta, (2008) A Netflow distance between labeled graphs applications in chemoinformatics. www.cs.berkeley.edu.

[25] O. Spjuth, E.L. Willighagen, R. Guha, M. Eklund, J.E.S. Wikberg, (2010) Toward interoperable and reproducible QSAR analysis: Exchange of data sets. Journal of cheminformatics, 2:5.

[26] Pradeep Chouksey, R.S. Thakur & R.C. Jain. (2009) *Exploring Interesting Multi-level patterns using graph based approach*, In Proc. 4th Indian International Conference on Artificial Intelligence (IICAI-09) pp 377-387.

[27] Han J.and Fu Y. (1995) Discovery of multiple-level association rules from large databases. In *Proc. 21st VLDB*, Sept. pages: 420-431.

[28] Mannila H., Toivonen H., and Verkamo A. (1994) Improved methods for finding association rules. In *Proc. AAAI Workshop on Knowledge Discovery*, pages 181-192.

[29] C.K.S. Leung, L.V.S. Lakshmanan, and R.T. Ng. (2002) Efficient dynamic mining of constrained using FP-trees. SIGKDD Explorations, 4(1), pp. 40-49.

[30] J.Pei, J. Han, and R.Mao. (2000) CLOSET: an efficient algorithm for mining frequent closed itemsets. In Proc. DMKD, pp. 21-30.

[31] Mannila H. and Toivonen H. (1995) Discovering frequent episodes in sequences. In *Proc. 1st International Conference on Knowledge Discovery and Data Mining (KDD),* pages 210-215.

[32] Mannila H.and Toivonen H. (1996) Discovering generalized episodes using minimal occurrences. In *Proc. 2nd International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 146-151.

[33] N.Deo, (2006) Graph Theory with Applications to Engineering and Computer Science, PHI New Delhi.

[34] Anurag Choubey, Dr. Ravindra Patel and Dr. J.L. Rana. (2011) Concurrent Edge Prevision and Rear Edge Pruning Approach for Frequent Closed Itemset Mining, International Journal of Advance Computer Science and Applications, Vol. 2, No. 11, pp. 65-70.

[35] L. Schietget, F. Costa, J. Ramon, L.D. Raedt, (2009) Maximum common subgraph mining: A Fast and effective Approach towards feature generation. In Proc. At SRL-MLG-ILP, Leven, Belgium,.

[36] C. Jiang, F Coenen, M. Zito, M. (2010) Frequent Sub-graph mining on Edge Weighted Graphs.

Author No. 1 - Anurag Choubey,

He has completed B.Sc. (Electronics) in 1990 from Dr. Hari Singh Gaur Vishwavidyalaya, Sagar, M.P., India and M.Sc. (Applied Physics) in 1993 from Govt. Engineering College, Jabalpur, M.P., India.
He has completed MCA in 2008 from Guru Ghasidas Vishwavidyalaya (Presently central University), Bilaspur C.G., India.
He has worked as a Lecturer at Govt. Engg. College, Jabalpur and Hitkarni College of Engineering & Technology, Jabalpur from 1998 to 2000. From October 2000 onwards he has been working in Technocrats Institute of Technology, Bhopal.
Currently working as Dean Academic, Technocrats Institute of Technology, Bhopal M.P., India.
He posses more than 13 years of experience in teaching and has worked in different capacities like controller of exam, admission inchagre and other administrative post.
His field of interest in research is Data Mining.

Author No. 2 - Dr. Ravindra Patel,
Associate Professor and Head, Department of Computer Applications at Rajiv Gandhi, Technological University, Bhopal, India. email: ravindra@rgtu.net

He has awarded Ph.D. degree in Computer Science. He posses more than 10 years of experience in teaching
post-graduate classes. He has published more than 15 papers in international and national journals and conference proceedings. He is member of International Association of Computer Science and Information Technology (IACSIT).

Author No. 3 - Dr. J.L. Rana,
Former Professor and Head, Department, of Computer Science and Engineering & Infromation Technology at MANIT, Bhopal, email: jl_rana@yahoo.com

India with 32 years of vast experience of teaching, out of which 19 years as Professor of Computer Science & Engg. and Information Technology
Currently working as Group Director, Radha Raman Group of Institute, Bhopal
He has completed B.E. (Hons) in 1968 from GEC, Jabalpur and M.S. Computer Control from University of Hawaii (UH), USA in 1972.
He has awarded Ph.D. degree in Computer from Indian Institute of Technology, Bombay in 1987.
He also posses more than 25 years of experience in post graduate teaching
He has published more than 30 papers in international and national journals and 65 international and national conference proceedings.
He has guided 12 nos. of Ph.D. and 5 no. in progress
He is a senior life member of CSI and Chairman CSI Bhopal Chapter for two terms.