A Novel Data Mining Method to Find the Frequent Patterns from Predefined Itemsets in Huge Dataset Using TM-PIFPMM

Saravanan.Suba[1] and Christopher.T[2]
[1]Research Scholar, Manonmaniam Sundaranar University, Tirunelveli, TN, India-627012
Email:saravanansuba@rediffmail.com
[2]Department of Computer Science, Government Arts College, Coimbatore, TN, India-641018
Email:chris.hodcs@gmail.com

Abstract-Association rule mining is one of the important data mining techniques. It finds correlations among attributes in huge dataset. Those correlations are used to improve the strategy of the future business. The core process of association rule mining is to find the frequent patterns (itemsets) in huge dataset. Countless algorithms are available in the literature to find the frequent itemsets. Most of the algorithms introduced in the literature finds all frequent itemsets for a given specified minimum support value. But in rare occasion, it is needed to check the occurrence of some predefined few frequent patterns in large dataset to improve the strategy of the future business. For this purpose, we previously introduced SIFPMM (Selective Itemsets Frequent Pattern Mining Method) method. FP-tree is one of the important methods for finding frequent patterns using two database scans. So this proposed TM-PIFPMM (Transaction Merging – Predefined Itemsets Frequent Pattern Mining Method) finds frequent patterns from predefined frequent itemsets using one database scan and it is compared with FP-tree and SIFPMM. The practical study of TM-PIFPMM proves that this method outperforms than FP-tree and SIFPMM.

Keywords-Apriori, FP-tree, SIFPMM, TM-PIFPMM, Minimum Support

I. INTRODUCTION

As with the invention of IT technologies, the amount of accumulation of data is constantly increasing. So it stores large volume of data in secondary storage. Thus the Data mining approaches come into picture to explore and analyse the database to find the exciting hidden patterns. So the data mining is motivated as decision support problems for most business organizations and is described as core area of research [1]. It has recently attracted significant care from database professionals, because of its applicability in numerous areas such as decision support, banking, insurance, retail, fraud detection, market strategy and financial forecasts. Later it has been implemented in pharmaceutics, health, government and all sorts of e-businesses [2].

A transaction in database usually holds the transaction id, transaction date and the items bought in the transaction. Any enterprise has commenced to identify that the information collected over years is an important strategic advantage and it also recognizes that there are prospective intelligences secreted in the enormous amount of data. So it needs techniques to mine the most valued information from warehoused data [3], [4]. The data mining contributes such methods to find useful unknown information. Data mining is a group of approaches for effective automated finding of formerly unknown, valid, novel, prized and clear pattern in huge database [4],[5].

Data mining tasks deal with the kind of patterns that can be mined. On the basis of the kind of data to be mined, there are two categories of tasks involved in Data Mining such as descriptive and predictive. The descriptive function mines with the general properties of data in the database. The descriptive techniques comprise of tasks like clustering, association and sequential mining. Predictive data mining jobs are those that do implication on input data to achieve at hidden knowledge and create exciting and useful estimate. The predictive mining methodologies include jobs like classification, regression and deviation.

Essential research areas in data mining are performance, mining approach, user interactions and data diversity. So the data mining approaches must be skilled and scalable well to the size of database and their execution times [6][7][8]. Association rule mining is a famous descriptive data mining techniques [7]. Since its introduction [9], association rule mining has advanced into one of the central data mining tasks and has involved notable attention among data mining researchers and specialists [10]. So this is a good method for finding correlations between variables in big database. For example, it has to find out how many of customers buy bread and jam together. Domain professional can utilize this detail for detecting the customer buying behaviours to maximize

the profit of the organization. So the core problem of association rule mining is frequent itemsets mining. The correlation rule of above said problem can be written as

$$\forall x \in customer, buys(x,\text{bread}) \rightarrow buys(x, jam)$$

Where x is a variable and buy(x, y) is a predicate that defines that consumer x buy item y. This rule specifies that a high percentage of people who purchase bread also buy jam [11]. Association rule mining can be described as follows. Let I = $\{i_1, i_2, \ldots, i_m\}$ is a set of items. A non-empty subset of I is called itemset and it is made as X= $\{i_1, i_2, \ldots, i_n\}$. Let D = $\{t_1, t_2, \ldots, t_k\}$ be a set of tuples. Each tuple T is a set of items such that T $\subseteq$ I. The total number of items in T is called size of the itemset and an itemset of size L is referred as L-itemset [12].

Let R, S be a set of items, Association rule has the form

$$R \rightarrow S \; and \; R \wedge S = \emptyset$$

Where R is an antecedent and S is the consequent of the rule. It applies two statistical methods that control the activity of association rule mining is support and confidence [4]. Firstly, it describes frequent itemsets based on least support threshold. After that, it uses least confidence to determine correlation between frequent itemsets. The support and confidence can be written as equations as follows [13].

$$Support\ (R \rightarrow S) = \sum(R \cup S)/N$$

$$Confidence\ (R \rightarrow S) = \sum (R \cup S)/ \sum R$$

Where, N denotes the number of transactions in D.

Number of researches have been introduced [1], [2], [3], [6], [7], [10], [11], [12], [13], [14], [15], [16], [17], [18], [19], [20], [21], [22], [23], [24] [25] in evolving competent method for finding frequent patterns after introducing Apriori by Agrawal et al. [9]. Among those techniques, FPTree [17] is one of the important and commonly used techniques for finding frequent itemsets. So the SIFPMM [25] and FPTree [17] are the important algorithms to prove the performance of this proposed TM-PIFPMM. This paper presents the TM-PIFPMM method to find significant frequent itemsets with less computing time than FP-tree and SIFPMM.

The rest of the paper is prearranged as follows: Related works are explained in section 2. The proposed method is debated in section 3. Experimental results and discussions are given in section 4. The conclusions and the ideas for future enhancements are written in section 5.

II. RELATED WORKS

The core task of association rule mining is to find the frequent itemsets from large database. It is very useful in market basket analysis. So many methods are introduced in the literature to find frequent itemsets. Usually all of them can be categorized into two types such as candidate generation [9] and pattern growth [17].

The very first algorithm was introduced for finding frequent itemsets is the AIS (Agrawal, Imielinski and Swami) algorithm presented by Agrawal et al. [9] which uses candidate generation technique. So it is the forerunner of all the methods to discover the frequent itemsets and confident association rules. The name of this algorithm was renamed as Apriori by Agrawal et al. [3], [19]. Several algorithms were introduced to improve the efficiency of Apriori. But Apriori algorithm regrets from many numbers of database scans necessary to find the frequent itemsets and take more time if the dataset size is enlarged [20].

In 2000 Han proposed a new algorithm named as FP-tree which represents pattern growth method and it uses FP-tree data structure. It finds frequent itemsets using two database scans by constructing and using FP-tree. If the database is very large, the construction of FP-tree is very difficult because the full FP-tree should be maintained in main memory until all necessary frequent itemsets to be found. So it suffers from the time required to build the FP-Tree structure for huge database. The rise in the size of the FP -tree with respect to the growth of database leads to difficult in making, search and insert operation on bulky FP-tree [17].

The SIFPMM [25] was introduced by us to find the frequent patterns from important frequent itemsets given by domain experts to improve the strategy of the future business. It works better than Apriori and FP-tree. Even though it works better than FP-tree for specified constrained frequent pattern mining, it further needs proficient algorithm with customized data structures to catch timely outcomes from ever growing database. So this paper introduces the TM-PIFPMM technique to find significant frequent itemsets from specified important frequent itemsets so that to decrease computing time than SIFPMM.

III. PROPOSED APPROACH

A. Dataset Size Reduction

Usually the dataset for finding frequent itemset contains identical transactions. Those identical transactions are merged as single transaction with the count for number of transaction merged [26]. This action decreases the total number of transactions in dataset as less than or equal to 2^I -1 transactions where I denotes the total number of different items in the shop. So this significantly reduces computing time of discovering frequent itemset.

B. Selection of Predefined Itemsets

Let K= {K_1, K_2… Km} be the set of frequent patterns found last time for the future strategic decision and L= {L_1, L2… Ln} be set of patterns collected from K based on condition stated by the proficient domain expert for finding the presence of its frequency in current large dataset to decide the future profit of the enterprise. This can be mathematically said in tuple relational calculus as

$$\{L|ConditionOn(K)\}$$

L comprises set of all patterns which fulfils the domain expert conditions on K to improve the future business strategy. Usually those patterns are caught early by domain expert and saved in the text file before executing this proposed method.

C. Occurrence Count Table

This algorithm apply one table that's name is Occurrence Count Table (OCT). It has two fields such as predefined patterns and occurrence count value. This table holds entries for all patterns in L and frequency count of each pattern that are identified in transaction database. The frequency count of each pattern is the count of the occurrence of such itemset in transactional database D. This table is formed and may be retained in the memory till the specified frequent patterns are not found [21]. The format of Occurrence Count Table (OCT) is shown in table 1

TABLE 1: Structure of Occurrence Count Table (OCT)

Sl. No.	Predefined Patterns	Occurrence Count(OC)
1		
.		
.		
N		

D. Proposed Algorithm

1. Algorithm: The TM-PIFPMM
2. Input: A database D, Minimum Support Value, predefined patterns L
3. Output: The frequent patterns F
4. begin
5. build TMT (Transaction Merge Table);
6. read the database D record by record until it reaches the end of record
7. {
8. update the TMT;
9. count the number of different items involved in the data base;
10. }

/* Construction of OCT*/

11. for each $t_i \in L$
12. {
13. OCT_PP(i)←t_i
14. OCT_OC(i) ← 0
15. }
16. F←{ ϕ }
17. for each $X_i \in$TMT
18. {
19. for each $X_j \in$ OCT
20. {
21. If ($X_j \subseteq X_i$)
22. {
23.OCT_OC(j)=OCT_OC(j)+1
24. }
25. }

26. }
27. for each $X_j \in OCT$
28. {
29. if OCT_OC(j) >=MSV
30. {
31. $F \leftarrow \{ F \cup X_j \}$
32. }
33. }
34. End

E. Illustration of Proposed Technique

Let it Consider the transaction set D with 10 transactions, minimum support value as 4 and items I={X, Y, Z}. The transaction set is shown in table 2

TABLE 2: Transactional Database

Tid	Itemset
T1	X,Y,Z
T2	X,Y
T3	Y,Z
T4	X
T5	X
T6	X,Y,Z
T7	Y,Z
T8	Y,Z
T9	X,Y
T10	X,Y,Z

The predefined patterns to find its occurrence in the above database is given in table 3

TABLE 3: Predefined Patterns

Predefined Patterns
X,Y
X,Z
Y,Z

The algorithm scans the transactions one by one and merges the identical transactions and store those merged transactions into a table called TMT as shown in table 4

TABLE 4: Transaction Merging Table (TMT)

Sl. No.	Itemset	Count
1	X,Y,Z	3
2	X,Y	2
3	Y,Z	3
4	X	2

Next step is to construct the OCT with initial values as shown in table 5

TABLE 5: Initial Occurrence Count Table

Sl. No.	Patterns	Occurrence Count
1	X,Y	0
2	X,Z	0
3	Y,Z	0

TABLE 6: Updated Occurrence Count Table

Sl. No.	Patterns	Occurrence Count
1	X,Y	5
2	X,Z	3
3	Y,Z	6

It is witnessed from table 6 that the frequent patterns discovered from the given set of predefined patterns are F= {X, Y}, {Y, Z}}.

IV. EXPERIMENTS

A. Experiments on Synthetic Datasets

Several experiments were done to evaluate the performance of the proposed method. The intel® core ™ i5-2450m CPU @2.5 GHZ, 4.0GB RAM ,64bit windows 7 operating system and NetBeans IDE 8.0.2 were used to execute the experiments. The synthetic dataset of 2000, 6000, 11000 and 22000 with 10 items and 8 selective patterns were created to check the scalability of proposed TM-PIFPMM with implemented version of FP-tree [26] and SIFPMM.

The first experiment was done by applying the above specified four groups of dataset in FP-tree, SIFPMM and TM-PIFPMM with 5% minimum support value. The corresponding execution time is shown in Table.7.

TABLE 7: Execution Time Comparison for a Few Datasets

Sl. No.	No. of Transactions	Execution Time in Milliseconds		
		FP-tree	SIFPMM	TM-PIFPMM
1	2000	150	80	42
2	6000	289	120	45
3	11000	330	145	48
4	22000	450	280	55

It is seen that the performance time is decreased linearly from FP-tree to SIFPMM to TM-PIFPMM and the differences continues even though the number of transactions increases.

The Fig.1 shows the performance of FP-tree, SIFPMM and TM-PIFPMM according to the run time of each method for given four datasets. It clearly demonstrates that the TM-PIFPMM outperforms FP-tree and SIFPMM.

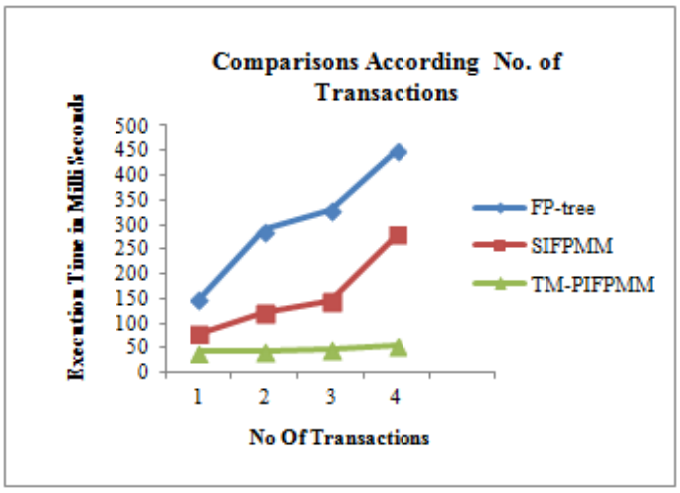

Fig.1: Time Comparisons for a Few Datasets

The table 8 indicates the execution time to find significant frequent patterns produced by FP-tree, SIFPMM and TM-PIFPMM for different support values such as 1%, 5%, 10% and 20% for 11000 transactions.

TABLE 8: Execution Time Comparisons for Different Thresholds

Sl. No.	Minimum Support Threshold	Execution Time in Milliseconds		
		FP-tree	SIFPMM	TM-PIFPMM
1	1	298	150	48
2	5	288	148	47
3	10	237	147	47
4	20	205	144	46

The table 8 clearly shows that the implementation time is reduced from FP-tree to SIFPMM to TM-PIFPMM. So TM-PIFPMM takes less computing time compared to FP-tree and SIFPMM, though it varies the support threshold value.

The Fig.2 exhibits the performance of FP-tree, SIFPMM and TM-PIFPMM according to the implementation time for 4 different minimum support thresholds with 11000 transactions. It obviously clarifies that the TM-PIFPMM outperforms than FP-tree and SIFPMM.

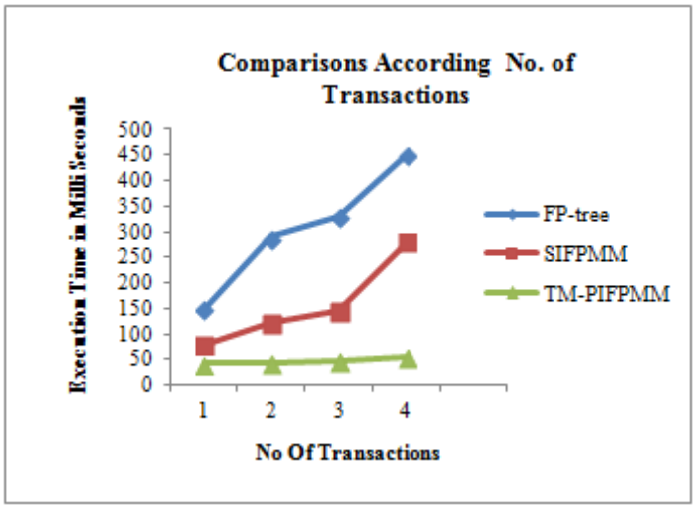

Fig.2. Response Time Comparison of Different Thresholds

B. Performance Analysis of Proposed Method

It can be clearly seen that the execution time taken by TM-PIFPMM is very small than the FP-tree and SIFPMM with the help of Fig.1 and Fig.2. In our previous work, it was proved that the SIFPMM works better than FP-tree [25]. So the performance of this proposed method is compared with SIFPMM.

The table 9 shows that the percentage of time reduction of using TM-PIFPMM against SIFPMM for different four group of synthetically generated transactions.

Table 9: Time Reduction Rate with Different Datasets (SIFPMM & TM-PIFPMM)

Sl. No.	Number Of Transactions	Execution Time In (Ms)		% Of Time Reduction
		SIFPMM	TM-PIFPMM	
1	2000	80	42	47.50
2	6000	120	45	62.50
3	11000	145	48	68.96
4	22000	280	55	80.35

The Fig.3 shows that the time reduction rate increases as number of transactions increases and it strongly proves that the use of TM-PIFPMM against SIFPMM will take less execution time. The average time reduction rate for using TM-PIFPMM against SIFPMM of this case is 64.82 %.

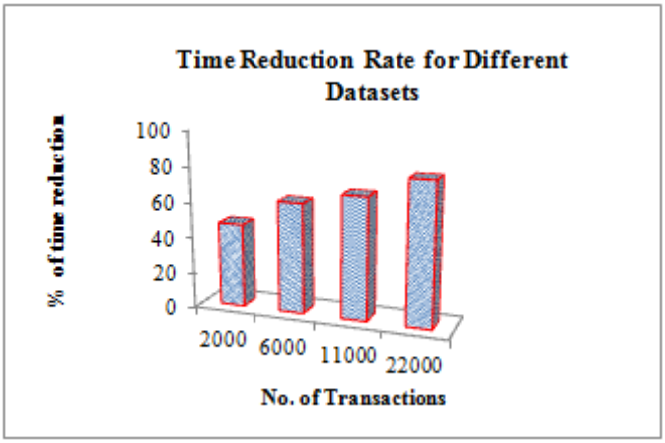

Fig.3. Time Reduction Rate of Using TM-PIFPMM against SIPMM for Different Group of Transactions

The table 10 displays that the percentage of time reduction of using TM-PIFPMM against SIFPMM for groups of support thresholds for synthetically generated 11,000 transactions.

TABLE 10: Time Reduction Rate of Using TM-PIFPMM against SIFPMM with Different Threshold Levels for 11,000 Records

Sl. No.	Minimum Support Threshold	Execution Time In Milliseconds		% Of Time Reduction
		SIFPMM	TM-PIFPMM	
1	1	180	48	73.33
2	5	148	48	68.24
3	10	147	47	68.02
4	20	144	46	68.05

The Fig.4 illustrates the time reduction rate of using TM-PIFPMM against SIFPMM for four levels of support thresholds. It also indicates that the time reduction rate is approximately equal, even though the support values are changed. The average time reduction rate for using TM-PIFPMM against SIFPMM of this case is 69.41%.

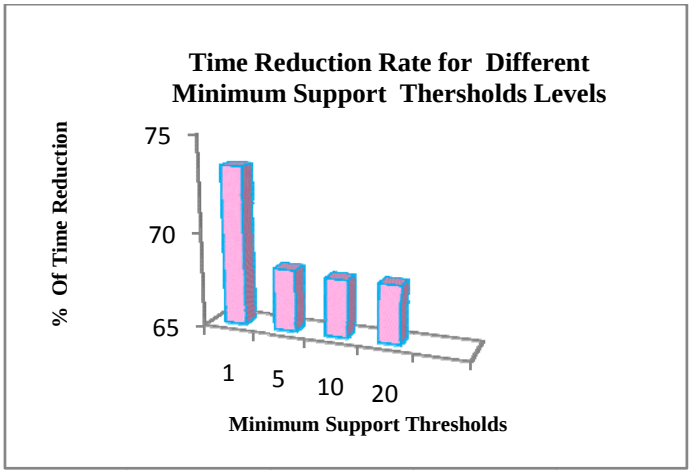

Fig.4. Time Reduction Rate of Using TM-PIFPMM against FP-Tree for Different Support Thresholds on 11000 Synthetic Dataset

V.CONCLUSIONS AND FUTURE GROWTHS

Mining association rule is one of the important descriptive data mining algorithms to find correlation in large dataset to improve the future strategy of the business to maximize profit of any enterprise. Frequent pattern mining is the core step of find correlation among attributes. This TM-PIFPMM finds frequent patterns from the important itemsets recommended by domain experts to make correct decisions to improve the future business. The experimentation results proves that the proposed methodology takes less execution times than FP-tree and SIFPMM even though it varies the number of transactions or least support threshold value on synthetically generated datasets. Hence the performance of TM-PIFPMM is better than FP-tree and SIFPMM for horizontal scalability of transactions and varied support thresholds. The suitable real time datasets should be examined in future to improve its proficiency further.

REFERENCES

[1] Ashok Savasere Edward Omiecinski Shamkant Navathe," An Efficient Algorithm for Mining Association Rules in Large Databases", Proceedings of the 21st VLDB Conference Zurich, Swizerland, 1995.
[2] Ya-Han Hu a, Yen-Liang Chen," Mining Association Rules with Multiple Minimum Supports: A New Mining Algorithm and A Support Tuning Mechanism", ELSEVIER Decision Support Systems Vol.42, page No.1 – 24, 2006.
[3] Rakesh Agrawal Ramakrishnan Srikant," Fast Algorithms for Mining Association Rules", Proceedings of the 20th VLDB Conference Santiago, Chile, 1994.
[4] G. K. Gupta," Introduction to Data mining with Case Studies", PHI learning private limited, New Delhi, 2009.
[5] Saravanan.Suba and Dr. Christopher.T, "A Study on Milestones of Association Rule Mining Algorithms in Large Databases" International Journal of Computer Applications (0975 – 888) Volume 47– No.3, June2012.
[6] Saravanan Suba, Christopher T ," An Efficient Data Mining Method To Find Frequent Itemsets In Large Database Using TR-FCTM ", ICTACT Journal On Soft Computing , Vol.06,Issue 02,January 2016.
[7] S.Shankar and T.Purusothaman, "Utility Sentient Frequent Itemset Mining and Association Rule Mining: A Literature survey and Comparative Study", International Journal of Soft Computing Applications ISSN: 1453-2277 Issue 4, pp.81-95, 2009.
[8] N.P.Gopalan and B.Sivaselvan, "Data mining Techniques and Trends", PHI Learning private limited, New Delhi, 2009.
[9] R.Agrawal, T.Imielinski, and A.Swami, "Mining Association Rules Between Sets of Items in Large Databases", In proceedings of the ACM SIGMOD International Conference on Management of data, pp. 207-216,1993 .
[10] M. J. Zaki and C.J. Hsiao, "CHARM: An Efficient Algorithm for Closed Association Rule Mining", Technical Report 99-10, Computer Science Dept., Rensselaer Polytechnic Institute, October 1999.
[11] Yin-Ling Cheung and Ada Wai-Chee Fu,"Mining Frequent Itemsets without Support Threshold: With and without Item Constraints", IEEE Transactions on Knowledge and Data Engineering, VOL. 16, NO.9, pp.1052-1069, September 2004.
[12] Claudia Marinica and Fabrice Guillet, "Knowledge-Based Interactive Post mining of Association Rules Using Ontologies", IEEE Transactions On Knowledge And Data Engineering, Vol. 22, No. 6, June 2010.
[13] Saravanan Suba, Christopher T, "DSMA Techniques For Finding Significant Patterns In Large Database", International Journal On Computer Science And Engineering,Vol.7 No.11 Nov 2015.
[14] Soo J, Chen, M.S, and Yu P.S, "Using a Hash- Based Method with Transaction Trimming and Database Scan Reduction for Mining Association Rules" , IEEE Transactions on Knowledge and Data Engineering, Vol.No.5, pp.813-825, 1997.
[15] Toivonen H, "Sampling large databases for association rules ", In VLDB Journal, pp. 134-145, 1996.
[16] Park, J. S, Chen, M.S and Yu P. S, "An Effective Hash Based Algorithm for Mining Association Rules" , In Proceedings of the 1995 ACM SIGMOD International Conference on Management of Data, M. J. Carey and D. A. Schneider, Eds. San Jose, California, pp.175-186, 1995.
[17] Jiawei Han, Jian Pei, and Yiwen Yin, " Mining Frequent Patterns without Candidate Generation", www.hanj.cs.illinois.edu/pdf/sigmod00.pdf.
[18] Mohammed Al-Maolegi1, Bassam Arkok," An Improved Apriori Algorithm for Association Rules", International Journal on Natural Language Computing (IJNLC) Vol. 3, No.1, 2014.
[19] Srikant.R, Agrawal.R, "Mining generalized association rules", VLDB '95 Proceedings of the 21th International Conference on Very Large Data Bases Morgan Kaufmann Publishers Inc. San Francisco, CA, USA, pp. 407-419, 1995.
[20] Sunil Kumar.S, Shyam Karanth.S, Akshay.K.C, Ananth Prabhu, Bharathraj Kumar.M, "Improved Apriori Algorithm Based on bottom up approach using Probability and Matrix" International Journal of Computer Science Issues, Vol. 9, Issue 2, No 3, 2012.

[21] Ramratan Ahirwal, Neelesh Kumar Kori and Dr.Y.K. Jain, "Improved Data mining approach to find Frequent Itemset Using Support count table" , International Journal of Emerging Trends & Technology in Computer Science, Volume 1, Issue 2, , July – August 2012.
[22] S.Brin, R.Motwani, J.D. Ullman, and S.Tsur, "Dynamic itemset counting and implication rules for market basket data", In Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD'97), pages 255-264, May 1997.
[23] J.Han, Y.Fu, " Mining Multiple Level Association Rules In Large Databases" IEEE Transactions On Knowledge And Data Engineering, Vol.11, No.5, pp.789-805,Sep/Oct.1999.
[24] Gurneet Kaur, "Improving The Efficiency Of Apriori Algorithm In Data Mining", International Journal On Science, Engineering And Technology, Vol. 02, Issue 05, June 2014.
[25] Saravanan Suba, Christopher T "An Efficient Frequent Pattern Mining Algorithm To Find The Existence Of K-Selective Interesting Patterns In Large Dataset Using SIFPMM", in International Journal of Applied Engineering Research ISSN 0973-4562 Volume 11, Number 7 (2016) pp 5038-5045.
[26] ESouleymane Zida, Philippe Fournier-Viger, Jerry Chun-Wei Lin, Cheng-Wei Wu, Vincent. Tseng, "EFIM:A Highly Efficient Algorithm for High-Utility Itemset Mining", *www.philippe-fournier-viger.com.*