IDENTIFYING FREQUENT ITEMSETS FROM DATABASE USING FREQUENT PATTERN MINING

S.D. Kamble[1], S.M. Kakde[2], P.V. Barekar[3], K.R. Singh[4], D.B. Bhoyar[5]
[1,3,4]Computer Science & Engineering, Yeshwantrao Chavan College of Engineering, Nagpur, MS, India
[2]Electronics Engineering, Yeshwantrao Chavan College of Engineering, Nagpur, MS, India
[4]Electronics and Telecommunication, Yeshwantrao Chavan College of Engineering, Nagpur, MS, India
shailesh_2kin@rediffmail.com

ABSTRACT

Now a day, a huge amount of data is generated. But, to handle that data andgetting important information from that huge amount of data is becoming a complex thing. In data mining there are various possible ways to mine data. Among those ways the algorithms of association rule mining are important one. The basic algorithm of association rule miningI.e. Apriori algorithm has some limitations i.e. Number of candidate sets generation and multiple database scans which results huge memory consumption, etc. So, to do improvisation in the basic apriori algorithm for finding frequent itemsets the hybrid apriori algorithm is proposed and it combines the positive approaches of each weighted and HashTapriori algorithms. The benefits of both the algorithms reduce the candidate set generation with multiple scanning of database , optimization and data integrity are also taken care of which are all important in the data mining algorithms.Here, by using the transactional database for getting frequent itemsets with evaluating the computation time, space and accuracy of frequent itemsets and comparing the existing versions of apriori algorithm to our proposed approach.

Keyword *- Association Rule Mining; Frequent itemsets; Apriori Algorithm*

1. Introduction

With the fast development of IT companies are applying the data warehouse systems and using the data mining tools to detect and predict customer behavior. There are a number of successful companies which are earning more profits and higher values byusing data mining tools. For example, Tesco is one the most useful tool to analyze the transactions, develop strategies to each group of members and profile of their members in data mining. So, the data mining is a very useful term to gather important information from huge database.In data mining the association rule shows the correlations between the groups of objects in the database. It is used to showcase the interesting or relevant connection between a large numbers of data items. Beside this, it has thelarge number of data that continuouslycollected and stored in database. So, many industries areinterested in mining association rules from the larger databases. This process helps industries in decision making for the future use, such as cross-marketing, market-basket analysis and loss-leader analysis. For discovering association rules, there are two main steps shown in figure 1.

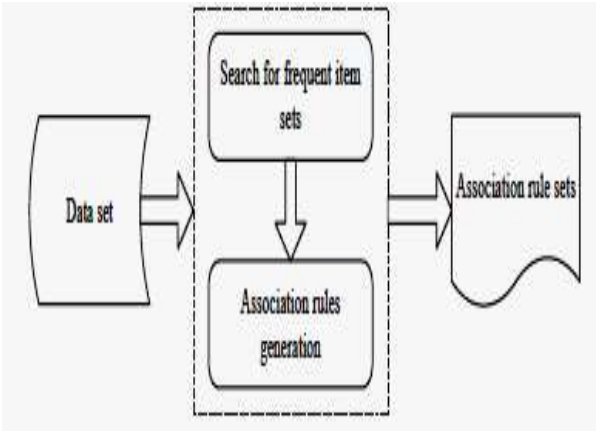

Figure 1: Steps of association rule mining

a. Finding the frequent itemsets with support count is greater than equal to the minimum support.
b. Generate the strong association rules from frequent itemsets. Association rules that satisfy minimum support and minimum confidence are strong association rules.
Association rule mining simply gives the set of itemsets (the transactions that each individual purchases in retail markets),apriori uses a "bottom up" approach for finding the frequent itemsets, where frequent subsets are generated at each pass called candidate sets and groups of 1-candidates are generated. The algorithm ends when there are no such extensions are found.

Support: A minimum **support** threshold in apriori algorithm is applied to find all frequent itemsets in a database.

$$Support(X \Rightarrow Y) = Support\ (X \cup Y) = P(X \cup Y)$$

Confidence: A minimum **confidence** constraint in apriori algorithm is applied to these frequent itemsetsin order to form rules.

$$Confidence\ (X \Rightarrow Y) = \frac{support\ (X \cup Y)}{support\ (X)} = P\left(\frac{Y}{X}\right)$$

In data mining, frequent itemsets are having the correlations among the number of fields of larger databases. The frequent mining is the concept of accepting items whose frequency of occurrence is equal or greater than a specific given minimum support. So, the many algorithms are introduced for frequent itemset and association rule mining. The apriori algorithm is used for association rule mining.
In the computer science and the data mining, the apriori is a classic algorithm for findingassociation rules. Apriori is designed to operate on databases with transactions (like, retail market transactions, shopping mall transactions, share market data, *etc.*). With the generation of apriori there are so many inventions of researchers that do improvisation in apriori algorithm. Apriori algorithm is abreadth-first search algorithm and having atree structure to count candidate itemsets efficiently. It generates the candidate itemsets of length k from itemsets of length $k - 1$. Then it prunes the candidates which have an infrequent itemsets. And according to the downward closure property, the candidate set are contains all frequent k-length itemsets. After that, it scans the transactions database and finds candidates sets among them.The remainder index of this paper is organized as follows: with describing introduction here, In Section 2 revised some existing approaches of apriori algorithms; Section 3 describes the work done on association rule mining. Section 4 shows the experimental results on transactional datasets with comparing existing approach with proposed one also in terms of computational time.Section 5 gives the conclusion on study of existing approaches with the proposed work and finally the paper ends with the future scope followed by references.

2. Apriori Algorithms

For the growth of business the frequent itemsets are very useful. This areimportant for business purpose like banks, stock markets,supermarkets, malls,etc., for finding frequent itemsets from algorithms of association rule mining like Apriori algorithm, FP-growth algorithm and improved versions of Apriori algorithms etc., this paper is mainly focuses on basic traditional Apriori algorithm.
Apriori algorithm require more database scans and generate the number of candidate sets. And it works on transactional databases. In every transaction there is set of items called itemsets. And it results frequent itemsets as support, confidence, strong rules and closed itemsets. Also it finds level-wise itemsets in the database. The Apriori algorithm is processed by two important steps like:

(1) **join**: making possible combinations of itemsets with their respective support counts and

(2) **Prune:**In this step check whether the value of support count is greater than or equal to the user-defined support threshold. Iftransaction is not found in database then delete that transaction from database otherwise, the transaction is added to database and perform the next steps to find frequent itemsets. Traditional apriori algorithm is very simple and clear algorithm in data mining. Joining and pruning are the most importantsteps in apriori algorithm to find frequent itemsets. But apriori algorithm also has some limitations.Section 4 is followed by the experimental results of the apriori algorithm.

2.1 Example

Apriori Algorithm is very basic algorithm in association rule mining for finding frequent itemsets. Among the transactional databases we get the frequent itemsets by applying two basic rules of apriori i.e., joining and pruning steps by generating possible combinations of itemsets we are able to join items and taking the items with greater than equal to supports value of items and eliminating other according to the minimum support value. To know its

working more take one basic example of apriori algorithm as follows:

Table 1 shows the simple database for apriori algorithm with TID as transactional id with their respective transactions. Each item corresponds to a goods such as "butter" or "bread".

Table 1: Database

TID	Items
100	1 3 4
200	2 3 5
300	1 2 3 5
400	2 5
500	1 3 5

In table 2, first pass build the list of frequent items from that data and count the frequencies i.e., support count of each item in the given database. For this example, let minimum support = 2. Therefore, the item which have less support count than minimum support are eliminated and rest are taken for the further pass so in this example item {4} has support count 1 so this is eliminated and the remaining one are the frequent items of first pass 1-candidate sets.

Table 2(C1, F1):Generating frequent 1-itemsets

C1

Item	Support
{1}	3
{2}	3
{3}	4
{4}	1
{5}	4

F1

Item	Support
{1}	3
{2}	3
{3}	4
{5}	4

In the next pass in table 3, it generate a list of 2-candidate of the frequent items. Here only take reference of previous pass frequent items. And make their respective 2-pair combinations where order doesn't matter only combination is required. And after joining do pruning on itemsets with minimum support = 2 so, according to it pruning of items is done. In this pass {1,2} itemset having the less count 1 so eliminate that one and take other itemsets as frequent itemsets of 2-candidate itemsets.

Table 3 (C2, F2): Generating frequent 2-itemsets

C2

Item	Support
{1,2}	1
{1,3}	3
{1,5}	2
{2,3}	2
{2,5}	3
{3,5}	3

F2

Item	Support
{1 3}	3
{1,5}	2
{2,3}	2
{2,5}	3
{3,5}	3

Table 4, in the third pass generates a list of all 3-candidate itemsets of the frequent items. In the example, contains 4 sets of 3-candidate sets frequent items but itemsets {1, 2, 3} and {1, 2, 5} support is smaller than our minimum support so here only 2 itemsets to be frequent in 3-pairs i.e., {1,3,5} and {2,3,5} with min support as 2 and having subsets of itemsets also be frequent.

Table 4 (C3, F3): Generating frequent 3-itemsets

C3

Item	Support
{1,2,3}	1
{1,2,5}	1
{1,3,5}	2
{2,3,5}	2

F3

Item	Support
{1,3,5}	2
{2,3,5}	2

In next pass build 4-pairs of frequent itemsets. So here we get {1, 2, 3, and 5} as 4-pair itemset as but with less count than minimum support so this is not the frequent itemsets. Therefore at last pass are obtained and it is considered being the output of applying apriori algorithm.At last, the final set of frequent itemsets are {1, 3, 5} and {2, 3, 5} with min support value as 2.

Limitations of Apriori Algorithm:

1) For generation of candidate sets, Apriori algorithm requires number of scans over the database.

2) Multiple scanning of databases takes lots of time to execute and increases I/Oload.

2.2 Hash-T Apriori Algorithm

HashTApriori algorithms implementation directly represents a hash table. This algorithm overcoming some of the drawbacks of traditional apriori algorithm by reducing the number of candidate k-itemsets. In the 2-itemsets is the key to improve performance of an algorithm. This algorithm uses a hash functions the hash value assigning to each individual item as per its occurrences to reduce the number of candidate itemsets.

This algorithm counts all the 1 candidate itemsets for each transaction at the same time. All the 2 candidate itemsets in the current transaction are store in a hash table. Moreover this algorithm uses a hash table to reduce the number of candidate itemsets. When the items of transactions satisfy the condition of minimum support then they generate the frequent itemsets. Hash based apriori algorithm generates the candidate itemsets same as the apriori algorithm.

Limitations of HashTApriori Algorithm*:*

1) Required more computational time.
2) Memory utilization is more
3) Requires more time to compute node processing.

2.3 Weighted Apriori Algorithm

In a basic apriori algorithm the number of frequent itemsets exist which have no meaning although they increase the database scans and require a lot of storage space. But, in weighted divide itemsets into number of categories and then assign values to those categories. Due to less number of categories, it is easy to set and adjust the weights. After dividing into the categories calculate the weighted supporting degree, and find the association rules from that who'sweighted supporting degrees to make effective marketing strategy using calculated weighted support and calculatedweighted confidence. The next step is like pruning and selection can make according to minimum weighted support and minimum weighted confidence. The same steps are following while calculating traditional apriorialgorithm but the only difference is that instead of all items groups are formed and algorithm is applied on that. Due to this the scan makes faster and less no of space is required as compared to traditional apriori algorithm. But, for making groups it requires a lot of analysis. If analysis made improper groups it would results into invalid association rules.

Retail markets are collect data and store them in the database. Divide the goods into number of categories according to the type of goods, respectively named A1, B1, C1, D1, each category has several kinds of goods, for example: A1 stands for drinks class, A11 indicates milk, A22 indicates Coke; B1 stands for food class, B11 indicates the bread, B22 indicates biscuits; C1 indicates fruit, C11 indicates apple.D1 stands for electrical class, D11, indicates rice cooker; Likewise all the goods are categorized and assigned their respective weights to them. And calculate the weighted supporting degree, for each frequent item of association rule.

Limitations of Weighted Apriori Algorithm*:*

1) Combinations of itemsets will be generated frequently and the candidate itemsets will be increases it.
2) For each transaction of weight computation it will take more time to execute
3) No dependency on data deviation

3. Work Done

In the proposed work methodology with taking the new concept, the existing work methodologies are reviewed and experienced to know it in more detail.The main approach of this proposed work is to combine the weighted value and HashTapriori algorithm to propose anew hybrid apriori algorithm. The figure 2 shows the flow of the proposed approach.

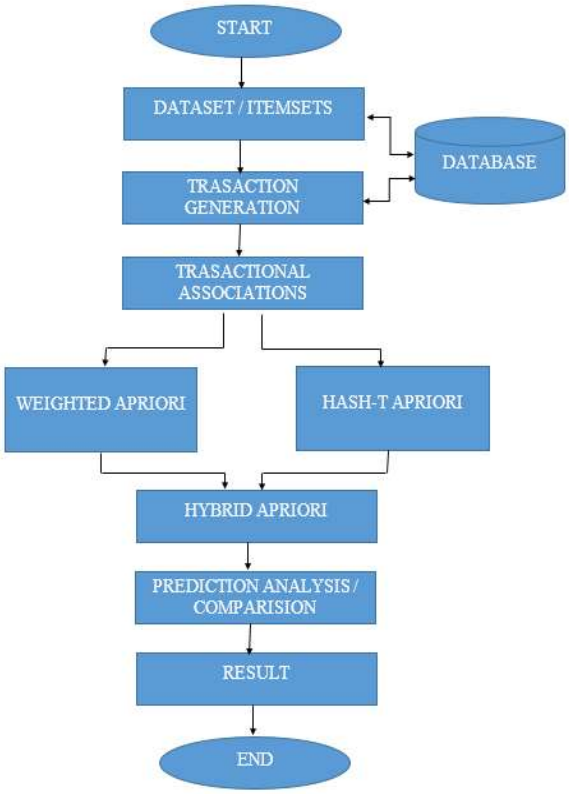

Figure 2: Flowchart of Proposed work

Firstly, Add items to the list and generate transactions from these items. There isalso one option of saving transactions in database so; it can be used in future. After generating transactions apply apriori algorithm on those transactions by providing minimum support and confidence as input to algorithm. From this the transactional associations frequent itemsets, closed itemsets, strong rules and maximal itemsets are generated. Frequent itemsets are those itemsets which repetitively come in number of transactions by satisfying minimum threshold condition. With this the strong rules are defined as those rules which satisfy the condition of minimum support and minimum confidence. The closed itemsets are those itemsets whose super proper itemsets do not exist and still those itemsets need to be frequent and closed are known as closed itemsets. Along with the closed itemsets maximal itemsets are those who do not have super itemsets.Through these transactional associations of apriori algorithm the weighted apriori and HashTapriori algorithm are calculated. For calculating weighted value itemsets mainly frequent itemsets are used to make it easier and in addition to this for giving parent child relationship to hash tree strong rules are used. The hash tree mainly uses the breadth first search for apriori with bottom up approach to find frequent itemsets steps in level wise manner. From the output sorting of both these algorithms one new hybrid apriori algorithm is proposed. Do the parallel executions of both versions of apriori algorithm i.e., weighted and HashT with the help of frequent itemsets and strong rule sets and at last do the prediction analysis and comparison of proposed approach to the existing ones.

In the initial phase the analysis of number of association mining algorithms especially apriori algorithm is start. In which it finds that the weighted and hash-t apriori perform well as compared to traditional apriori algorithm. The proposed work concentrates on deriving a new apriori algorithm which performs mining of frequent itemsets on frequent pattern mining for achieving computation of less memory usage and less computational time.When the proposed approach is implemented in a distributed environment it improves performance

The first design module of the proposed work includes the collection and study of relevant data set. For the said work the relevant data set is the record of daily transactions of supermarket. Hence the required data set is collected and studied successfully. Since the collection consists of clusters like milk, bread, butter, jam and pen pencil, sharpener, etc. On this sets the number of transactions are generated and savedin database. The analysis of apriori generates the result according to the applied methodologies.At the initial phase of proposed approach first add items to the item lists and make transactions from that. After generation of transactions make the transactional associations like support, confidence, closed itemsets and frequent itemsets. Now, applying the weighted value and HashT algorithm individually to get the efficient results, but individually this algorithm may not be more efficient to give the properoutputs. The positives approaches of both the algorithms are not able to archive efficient results,Because of that the new algorithm is invented with the benefits of both weighted apriori and HashTapriori algorithms. To overcome the limitations of both the algorithms a new hybrid approach is being introduced in the proposed work. So, the

hybridization of both these algorithm is essential. The combination of both these algorithm creates new hybrid apriori algorithm of better efficiency and it archives the frequent itemsets and reduces the computational time and space as well.

4. Experimental Results

While implementing algorithms to find frequent itemsets here retail market datasets are used. The retail market data comes in terms of transactions and to implement apriori algorithms and its said versions we also needs those transactional databases. While observing those transactions analyze the buying habits of customers and frequently bought itemsets. In this process it is observed that the existing apriori approaches consuming maximum computational time and memory space. So, it find that the existing apriori algorithms are not that efficient for the larger datasets. Hence there is a need of efficient algorithm which will consume less computational time and space to give frequent itemsets. Therefore a hybrid approach of apriori algorithm using some versions of apriori algorithms is proposed

.

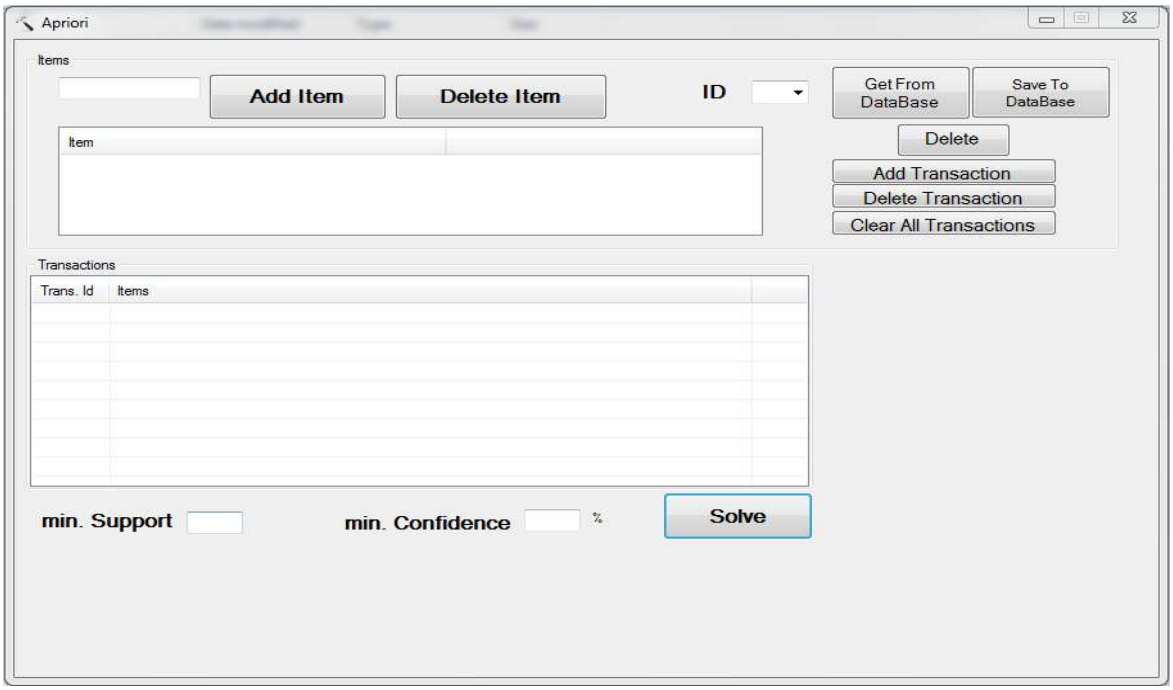

Figure 3: Main Form for Applying Apriori Algorithm

Figure 3. Simply illustrations the UI of generating transaction from itemsets. These UI facilities us to get items and generate transactions from which also those transactions can be saved in database and use in future. Even the transactions can be deleted and cleared. Also provide the minimum support and the minimum confidence to apriori in range of the percentage as input and solve button gives us the required output of apriori algorithm.

Figure 4, shows how to add items in list and generate transactions from those items in any order. Also here provide minimum support and minimum confidence for calculating frequent itemsets here provide 2 as minimum support and 60% for minimum confidence these values are given according to user but, as the values of minimum support and minimum confidence get frequent itemsets also change. Here we take a simple example of daily supermarket buying products like milk, bread, butter, etc. so from those items in transaction list generate their transactions. While for example support indicates that how many times the milk and bread was brought together and confidence shows the percentage of items if someone buys milk they will definitely buy bread so, displaying both of them together increases sell of bread more. These are the simple strategies of retailers to sales their products. While giving discount they also increase their sale. This undergoes the market basket analysis.

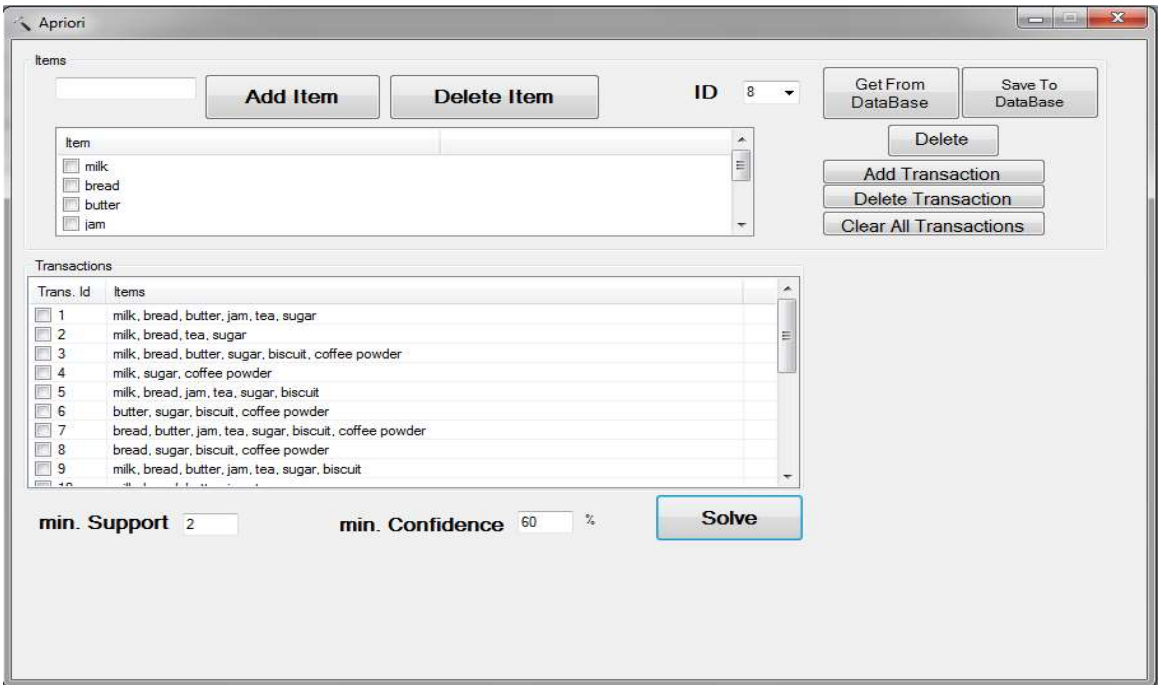

Figure 4: Adding Items and Generating Transactions

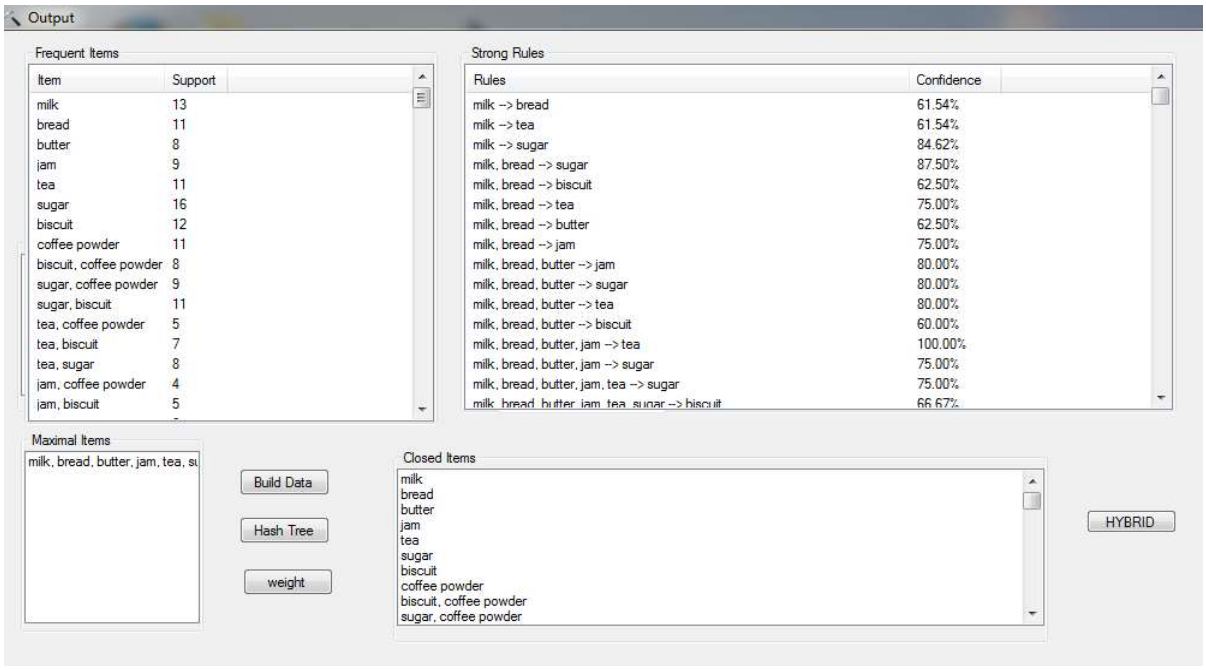

Figure 5: Generating Frequent Itemsets using Apriori Algorithm

Figure 5, Shows the output of apriori algorithm where frequent itemsets are obtained with their support values which are more than the minimum support. Also here calculated strong rules which generate possible combinations of frequent items with respective confidence values by satisfying minimum support and minimum confidence condition. Also in output get closed itemsets are obtained the itemsets that do not have any proper superset of frequent itemsets but they should be frequent and closed. And maximal items are that item which have no supersets and with maximum items in transactions so this is output of applying apriori algorithm on transactional datasets.

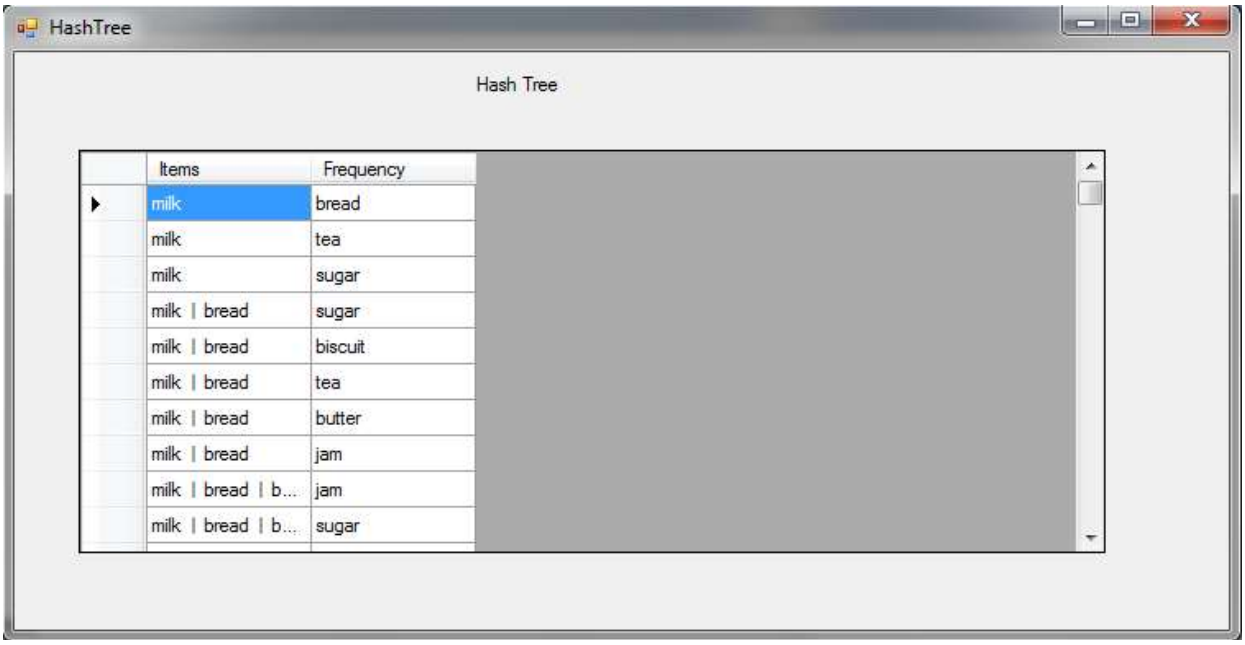

Figure 6: Generating Frequent Itemsets using HashTApriori Algorithm

HashTapriori algorithm mainly applies the hash function to store the items identity. Moreover, hash tree can be saved in tree bucket format by using hash tree HashT directly generates L2 I.e., 2-itemsets combination and generates tree then the execution of HashT is same as apriori here results belonging to only apriori algorithm should be taken. The output of HashTapriori is shown in figure 6, with parent and their child relation shown with the help of pipes. The parent child relation mainly shows the hashing link of itemsets which are frequently brought in specific manner.

Figure 7, shows the output of weighted apriori algorithm with their weights and items. In the weighted value algorithm firstly divide the items into the categories according to the item types and then on those groups assign the weights and then calculate weighted supporting degrees. Here, weighted value apriori algorithms are assigning the weights according to the positions of its items means according to the occurrences of the items coming in each transaction. Its execution is same as the apriori but only difference is that with the items there weight also be taken.

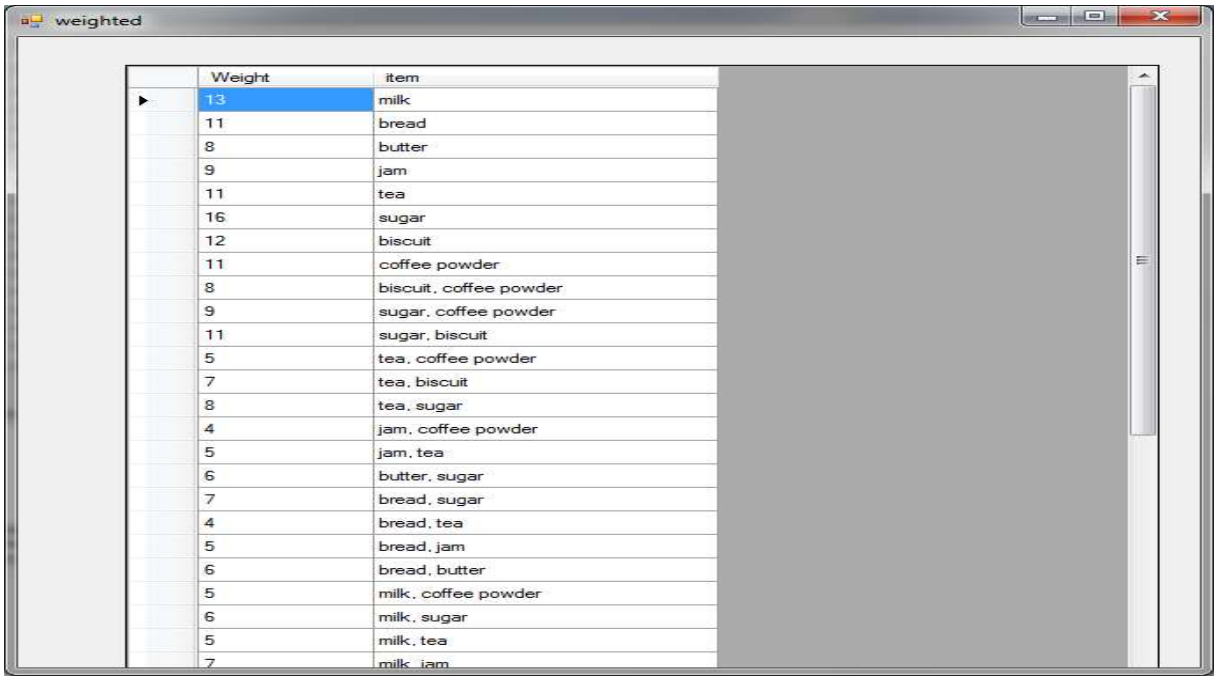

Figure 7: Generating Frequent Itemsets using Weighted Apriori Algorithm

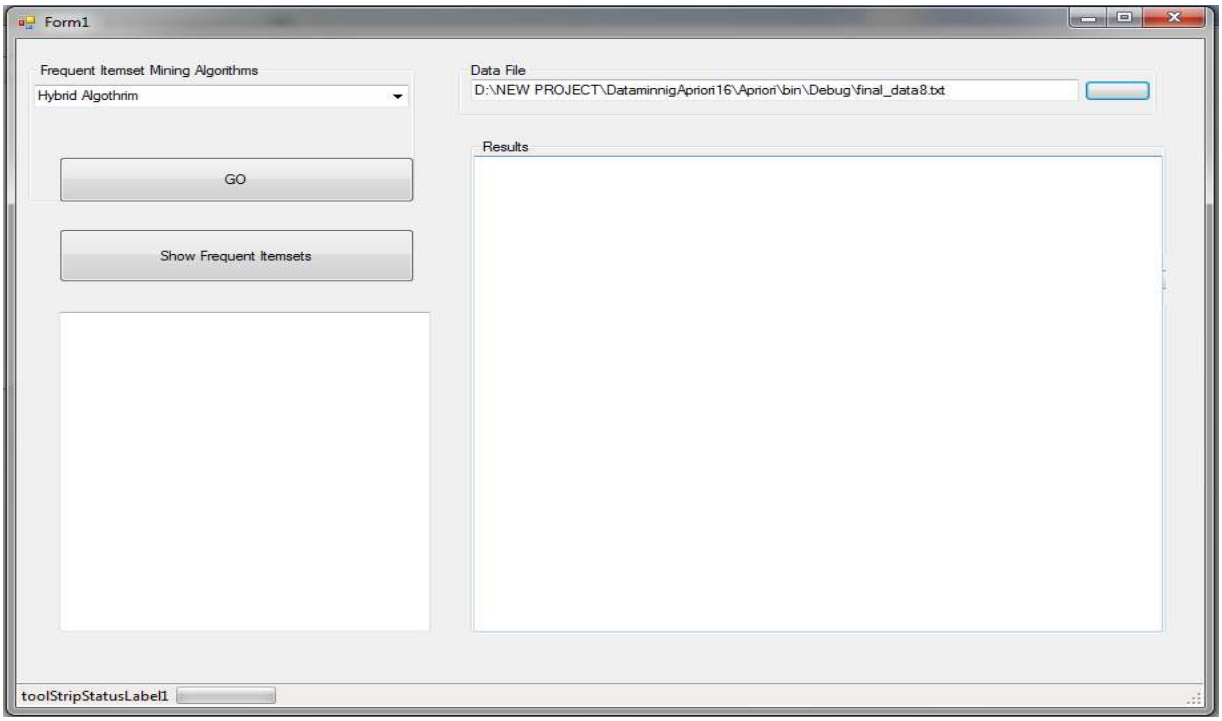

Figure 8: Hybrid Interface

Figure 8, is the hybrid apriori algorithms Interface where the hybridization is applied on the output of HashT and weighted apriori algorithm by applying frequent itemsets mining algorithm as hybrid Algorithm. Here the outputs files are taken in parallel way from this take weighted value from weighted apriori algorithm and hash tree value from HashT algorithm and comparing them in terms of frequent itemsets occurrences.

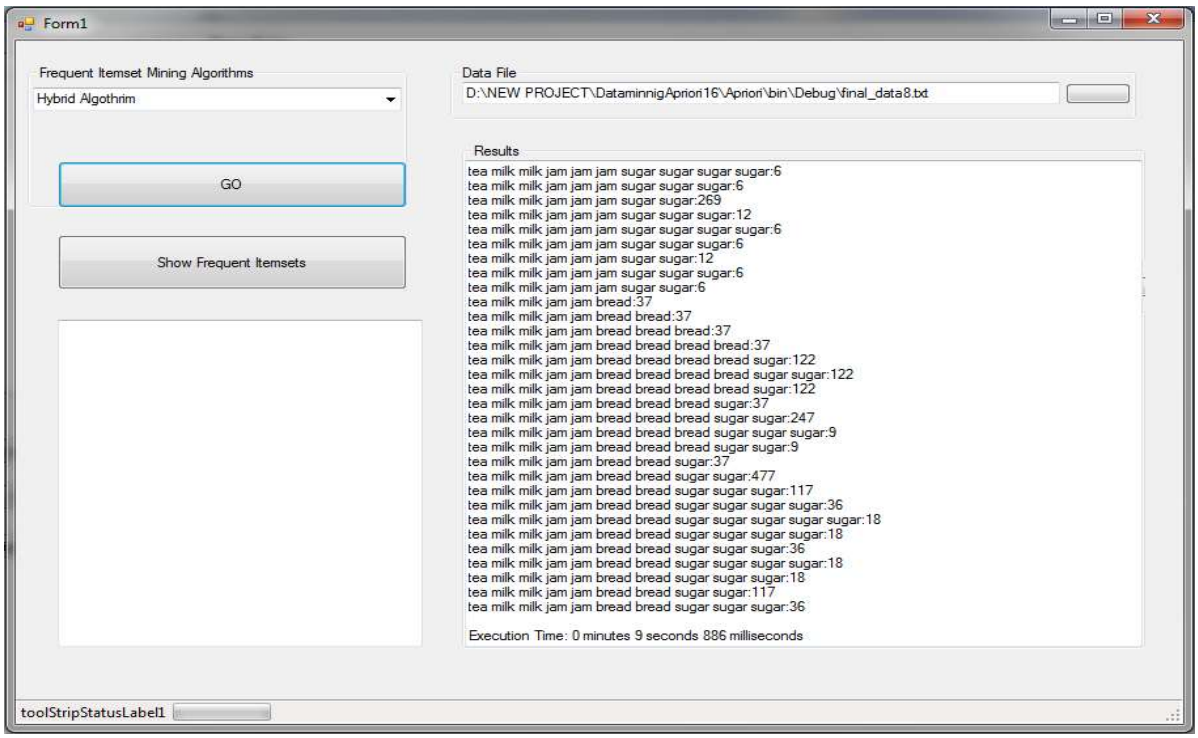

Figure 9: Data Analysis of Hybrid Algorithm

Figure 9, simply shows the analysis of transactions from both the HashT and weighted value apriori algorithm to execute the hybrid algorithm and get frequent itemsets from that algorithm with less computational time. With each analysis of frequent itemsets from both the algorithms in parallel way calculate their required time to complete each comparing

analysis in millisecond. This is a small transactional dataset where its time and analysis structure can be easily taken but for huge datasets it requires more time to analyze according to the size of itemsets and their transactions.

Figure 10, simply shows the output screen of hybrid apriori algorithm where application of hybrid apriori algorithm on the weighted value and HashTapriori algorithm gives frequent itemsets with less time than others according to the analysis output of HashT and weighted value apriori algorithm with its execution time.

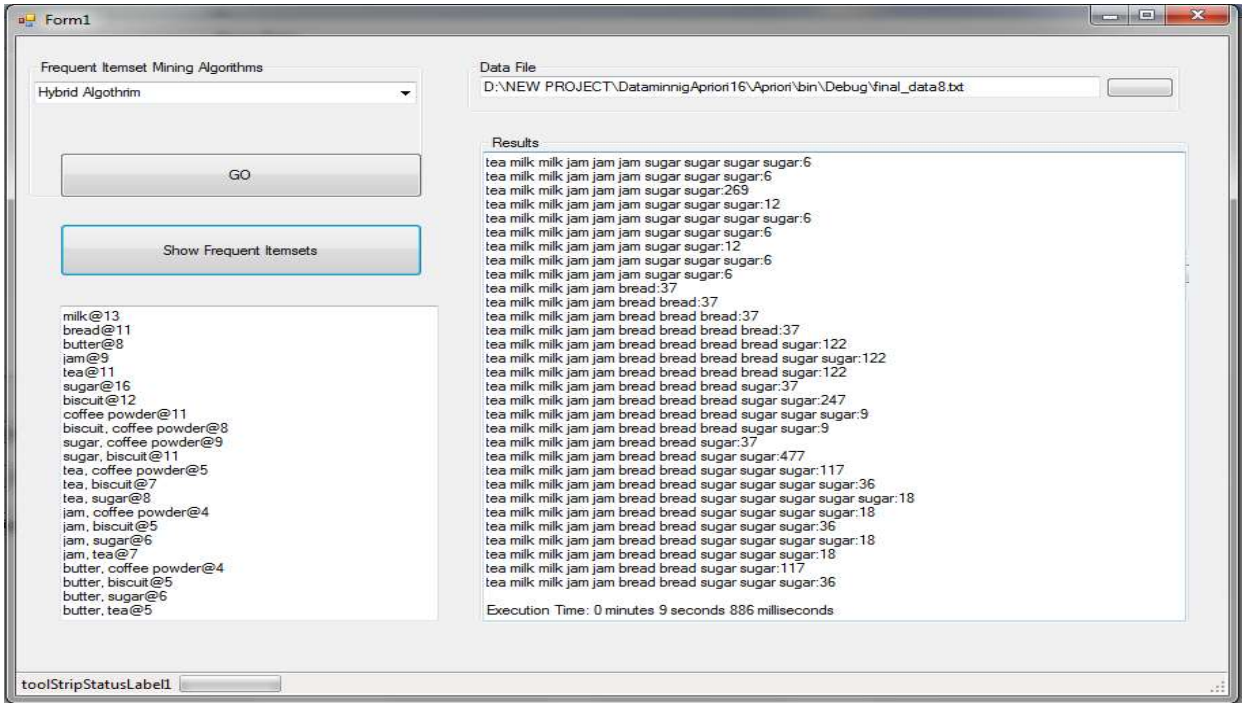

Figure 10: Final Transaction with time analysis

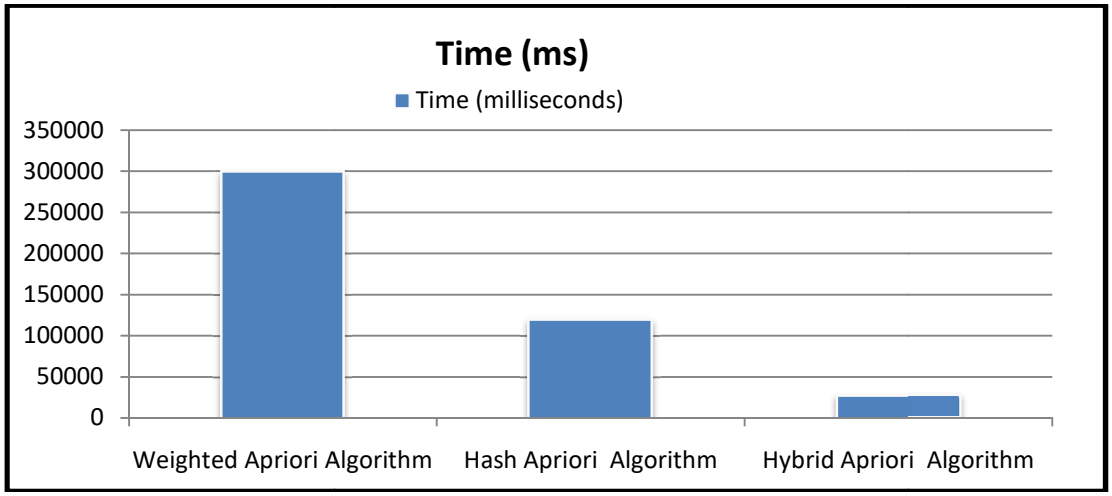

Figure 11: Time comparison of algorithms

A comparative approach among all the algorithms used in our research with respect to some parameters like, the new technique used in algorithms and how they best in nature also the Number of scans done in each algorithm respect to other and the storage structure used to store the items is given below in table 5. The best algorithm is decided according to these parameters. Also the figure 11 shows the time comparison between weighted value apriori, HashTapriori and hybrid apriori algorithm.

Table 5: Comparison of Apriori Algorithms

Attributes	Traditional Apriori Algorithm	HashTApriori Algorithm	Weighted Apriori Algorithm	Hybrid Apriori Algorithm
New Technique Used	Simple Apriori Candidate Generation approach	Hash structure introduced to quick retrieve data from database	Large amount of items divided categorized into groups	Sorted value of hash and sorted value of weighted
Number of scans	More than 1	1	More than 1	1
Storage structure used	Normal Database	Hash Table	Normal Database	Normal Database with Hash Table

5. Conclusion and Future Work

There are number of improvisations are possible in Apriori algorithm. But, the two main issues in apriori algorithm are number of candidate set generation and multiple timesdatabase scanning.There are many ways of improvements in the efficiency of apriori algorithm. From the overall observation, the weighted value Apriori and hash tree Apriori are the best and they give better efficiency in Apriori. But, individually they are not more efficient as they are together. Therefore, it is better to combineboth the algorithmsto getbenefits of them togetherto make more efficient. And, proposing one hybrid approach based on hash tree apriori together with weighted value apriori algorithm to improve computational time and memory usage of required algorithms.

In future, there is a scope for improvisation in existing apriori algorithms and can be applied on retail market, shopping mall, stock market databases to identify frequent itemsets from that with reducing database scan, generating less candidate sets on larger datasets. Also in future apply this hybrid algorithm can be applied on huge amount of data or say big data in unstructured format to show its nature and work on better time consumption of hybrid algorithm.

References

1. M. Chen, J. Han and P. Yu, "Data mining an overview from the database perspective," IEEE Transactions on Knowledge and Data Engineering, Volume 8, Issue 6, Pp.866-883, December 2009.
2. Y. Liu, "Study on Application of Apriori Algorithm in Data Mining", IEEEInternational Conference on Computer Modeling and Simulation, Pp.111-114,22-24 January 2010.
3. W. Hao-yu, J. Xiao-juan, X. Yun and L. Xing, "Applying Fast-Apriori Algorithm to Design Data Mining Engine",IEEEInternational Conference on System Science, Engineering Design and Manufacturing Informatization,Pp.63-65,12-14 November 2010.
4. R. Liang and J. Sun,"Frequent Items Mining Based on Weight in Data Stream", TENCON IEEE region 10 Conference, Pp.1-3,23-26 January 2009.
5. C. Song, "Research of Association Rule Algorithm Based On Data Mining," IEEE International Conference of Big Data Analytics (ICBDA), Pp.1- 4, 12-14 March2016.
6. O. Jamsheela and Raju.G, "Frequent Itemset Mining Algorithms: A Literature Survey," IEEE International Advance Computing Conference(IACC), Pp. 1099-1104, 12-13 June2015
7. L. Fang and Q. Qizhi, "The Study on the Application of Data Mining Based On Association Rules," IEEE International Conference on Communication Systems and Network Technologies (CSNT), Pp. 477-480, 11-13 May2012.
8. A. Singh and J. Agarwal, "Proposed Algorithm for Frequent Item Set Generation," IEEE International Conference OnContemporaryComputing (IC3), Pp.160-165, 7-9 August2014.

9. S. Patil and R. Deshmukh, "Review and Analysis of Apriori Algorithm for Association Rule Mining," IEEE International Journal of Latest Trends in Engineering and Technologies (IJLTET), Volume 6, Issue 4, March2016.
10. K. Rajeswari, "Improved Apriori Algorithm – A Comparative Study Using Different Objective Measures," IEEE International Journal of Computer Science and Information Technologies, Volume 6, Issue 3, 2015.
11. Y.Shaoqian, "A Kind of Improved Algorithm for Weighted Apriori and Application to Data Mining," IEEE 5th International Conference on Computer Science &Education (ICCSE), pp. 507-510, 24–27 August 2010.
12. M. Alharbill, S. Pathak and S. Rajasekaran, "Frequent Itemsets Mining on Weighted Uncertain Data," IEEE International Symposium on Signal Processing and Information Technology (ISSPIT), Pp. 000201- 000206, 15-17 December 2014.
13. A. Ehsan, and N. Patil, "Normalized Weighted and Reverse Weighted Correlation Based Apriori Algorithm," IEEE International Conference On Advance in Computing, Communication and Informatics (ICACCI), Pp. 841-847, 10-13 August2015.
14. J. Agarwal, and A. Singh, "Frequent Item Set Generation Based On Transaction Hashing," IEEE International Conference On Confluence the Next Generation Information Technology Summit (Conflence), Pp. 182-187, 25-26 September 2014.
15. Z. Zeng, H. Yang and T. Feng, "Using HMT and HASH_TREE to Optimize Apriori Algorithm," IEEE International Conference On Business Computing and Global Informatization, Pp. 412-415, 29-31 July 2011.
16. R.Rathinsabapathy, and R.Bhaskaran, "Performance Comparison Of Hashing Algorithm With Apriori," IEEE International Conference On Advances In Computing, Control, And Telecommunication Technologies, Pp. 729-733, 28-29 December 2009.
17. X. Geng and F. Tao, "A New Text Association Rule Algorithm Based On Concept Vector and Its Application," IEEE International Conference on Multimedia Information Networking and Security, Pp. 492-495, 2-4 November2012.
18. P. Bhandari, K. Rajeswari, S. Tonge, and M. Shindalkar, "Improved Apriori Algorithms – A Survey," IEEE International Journal of Advanced Computational Engineering and Networking, Volume-1, Issue- 2, April-2013
19. J. Deone and V. Jethan, "Frequent Patterns for Mining Association Rule in Improved Apriori Algorithm," IEEE International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), Volume 3, Issue 3, March2014.
20. H. Qiu-Yong, T. Ai-Long and S. Zi-Guang, "Optimization Algorithm of Association Rule Mining BasedOn Reducing the Time of Generating Candidate Itemset," IEEE International Conference On Automation & System Engineering (CASE), Pp. 1-4, 30-31 July 2011.
21. P. Mundra, A. K. Maurya and S. Singh, "Enhanced Mining Association Rule Algorithm with Reduced Time & Space Complexity," IEEE India Conference (INDICON), Pp. 1105-1110, 7-9 December 2012.
22. Sumangali.K, Aishwarya.R, Hemavathi. E and Niraimathi.A, "Mining Interesting Itemsets from Transactional Database," IEEE International Conference On Computational Intelligence and Computing Research (ICCICR), Pp. 1-4, 18-20 December 2014.
23. A. Singh, A. Kumar and A. Maurya, "An Empirical Analysis and Comparison of Apriori and FP- Growth Algorithm for Frequent Pattern Mining," IEEE International Conference On Advanced Communication Control and Computing Technologies (Lcaccct), Pp. 1599-1602, 8-10 May 2014.
24. X. Gu, X. Hou, A. Wang, H. Zhang, X. Wu and X. Wang, "Comparison and Improvement of Association Rule Mining Algorithm," IEEE International Computer Conference On Wavelet Active Media

Technology and Information Processing (ICCWAMTIP), Pp. 383-386, 18-20 December 2015.

25. P. Wang, C. hongand Lei Wang, "An Improved Algorithm for Mining Association Rule in Relational Database," IEEE International Conference on Machine Learning and Cybernetics, Pp. 247-252, 13-16 July 2014.
26. T. Singh and M. Sethi, "Sandwich-Apriori: A Combine Approach of Apriori and Reverse-Apriori," IEEE India Conference (INDICON), Pp. 1-4, 17-20 December 2015.