Proceedings of the Second International Conference on Research in Intelligent and Computing in Engineering pp. 187–192 DOI: 10.15439/2017R83
ACSIS, Vol. 10 ISSN 2300-5963

Evaluation of Apriori Algorithm on Retail Market Transactional Database to get Frequent Itemsets

Pooja R. Gaikwad[1], Shailesh D. Kamble[2], Nileshsingh V. Thakur[3], Akshay S. Patharkar[4]

[1]PG Scholar, Computer Science & Engineering, Yeshwantrao Chavan College of Engineering, India
[2]Computer Science & Engineering, Yeshwantrao Chavan College of Engineering, India
[3]Computer Science & Engineering, Prof Ram Meghe College of Engineering & Management, India
[4]Computer Technology, K.D.K. College of Engineering, India
[1]askpoojagaikwad@gmail.com, [2]shailesh_2kin@rediffmail.com, [3]thakurnisvis@rediffmail.com, [4]akshay.patharkar7@gmail.com

Abstract* – *In Data mining the concept of association rule mining (ARM) is used to identify the frequent itemsets from large datasets. It defines frequent pattern mining from larger datasets using Apriori algorithm & FP-growth algorithm. The Apriori algorithm is a classic traditional algorithm for the mining all frequent itemsets and association rules. But, the traditional Apriori algorithm have some limitations i.e. there are more candidate sets generation & huge memory consumption, etc. Still, there is a scope for improvement to modify the existing Apriori algorithm for identifying frequent itemsets with a focus on reducing the computational time and memory space. This paper presents the analysis of existing Apriori algorithms and results of the traditional Apriori algorithm. Experimentation carried out on transactional database i.e. retail market for getting frequent itemsets. The traditional Apriori algorithm is evaluated in terms of support and confidence of transactional itemsets.

Keyword - Frequent itemsets; Association Rule Mining; Frequent pattern mining; Apriori; FP-growth

I. INTRODUCTION

The tremendous growth of information technology within the companies, businesses and governments, has created immense Databases (DBs). This trend creates a prompt requirement for novel tools and techniques for intelligent DB analysis. i.e., we are drowning in information but starving for knowledge! These tools and techniques are the topics of the field called "data mining" or "Knowledge Discovery in Databases" (KDD). It is used for finding hidden and probably useful patterns and knowledge in databases. In which association rule mining is important factor with various algorithms. In association rule mining frequent itemsets gives the correlation between the larger datasets. Frequent itemset are the itemsets which are present mostly in the transactional databases. So, to get more profit from this we need to find the frequent items. The Apriori algorithm is used for the association rule mining to find frequent itemsets from datasets. Apriori is the basic algorithm for mining frequent itemsets in a set of transactions. A set of items is known as itemset. If the occurrence of items in a particular transaction is frequent, it is called as frequent itemset and the support of a frequent itemset is greater than some user-specified minimum support. Mainly to perform Apriori we have to give transactional database and minimum support value as input and frequent itemsets as output.

Association Rule Mining: Association rule mining is important aspect of data mining concepts which states as following rules, Let $I = (i_1, i_2 \ldots\ldots, i_m)$, be a group of items. Let D be a group of transactions, wherever all transaction T includes one or more set of items in I, such $T \subseteq I$. Every transaction is always comes with a unique identifier, called TID. Let X contains a group of items. A transaction T is claimed to have X only if $X \subseteq T$. An association rule defines its expression as a $X \Rightarrow Y$, wherever X and Y are nonempty item sets (i.e. $X \subseteq I, Y \subseteq I$). This rule is termed as antecedent, such that $X \cap Y = \emptyset$. The rule $X \Rightarrow Y$ contains the transaction set D with support s, wherever $s\%$ of transactions in D that holds $X \cup Y$. The rule $X \Rightarrow Y$ has confidence$c$, within the transaction set D, wherever $c\%$ of transactions in D have X that also have Y.

Support: The rule $X \Rightarrow Y$ has support s within the transaction set$D$, if this is the case of transactions in D contains $X \cup Y$. Rules that have a s greater than or equal to the user-defined support is given as a minimum support threshold(min_sup).

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

Confidence: The rule $X \Rightarrow Y$ has confidence c within the transaction set$D$, if recollect transactions in D contain X that also contain$Y$. Rules that have a c larger than or equal to a user-specified confidence is termed as a minimum confidence threshold (min_conf).

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

Therefore, the smaller support with greater the confidence values are used. Satisfying the rules of support and confidence the frequent itemsets are generated by the strong rules. Also the useful information found from datasets in terms of strong rule sets to find frequent itemsets. The issues generated in association rules are of two kinds: (1) Find all the items having greater support value of transaction than the minimum support. These are the frequent itemsets. And alternative

itemset referred to as infrequent itemsets; (2) Use the frequent itemsets to get the specified rules. Support and confidence are the important factors in the Apriori algorithm.

There's a large union between the literature that the primary sub problem is that the necessary of two algorithms. This is because it's more time consuming because of the enormous search space and therefore the rule generation section can be done in main memory in a very simple means once the frequent itemsets are found. That's the reason for the huge awareness researchers paid to the current problem within the recent years. The remainder of this paper is organized as follows: Section II describes the related work on frequent pattern mining. Section III discusses about the analysis of traditional Apriori algorithm. Section IV summarizes the experimental results on transactional datasets. The discussion about the content of this paper is described in section V. Section VI summarizes the conclusion based on the study of existing approaches and finally the paper ends with the future scope followed by references.

II. RELATED WORK ON FREQUENT PATTERN MINING

Literature available on frequent pattern mining is very large. This section presents the detail overview of existing techniques to mine large datasets efficiently. C. Song [1] defines association rule mining with algorithms of data mining performing on larger datasets and generates output efficiently. Jamsheela and Raju [2] compare both the Apriori and FP growth algorithms and based on the survey they suggested that the FP growth algorithm performs better than traditional Apriori algorithm. Fang and Qizhi [3] proposed an improved Apriori algorithm with better in performance than previous one and performing less scan with generating less number of candidate sets to improve algorithm efficiency. Singh and Agarwal [4] proposed the optimized algorithm i.e. FI generator with fewer numbers of database scans. Also reduces the number of candidate set iteration and pruning techniques for reducing storage space. Patil and Deshmukh [5] mainly focuses on the analysis and improvement of Apriori algorithms based on parallelization, time efficiency, interesting itemsets mining, stopping conditions, etc. Rajeswari [6] proposed modified version of Apriori algorithm and compare with different existing Apriori algorithms in terms of different evaluation parameters. Shaoqian [7] find the minimum support and confidence according to the user defined minimum support value with the pruning and joining concept of Apriori algorithm and generates frequent itemsets with greater efficiency. Alharbill et al. [8] proposed two algorithms called horizontal weighted uncertain Apriori (HWUAPRlORl) and Vertical weighted uncertain frequent itemset mining (VWUFIM) for mining frequent itemsets from the any type of larger databases. Ehsan and Patil [9] proposed an algorithm for finding frequent and infrequent itemsets based on Apriori algorithm. The two new approaches are introduced i.e. the normalized weighted and reverse weighted algorithm for finding frequent itemsets. Mainly the weighted algorithm concept forms the groups of similar itemsets and assigning the weights to them.

Agarwal and Singh [10] suggested that by using hash function with hash value generating a tree with Direct Hashing and Pruning (DHP), and Perfect Hashing and Pruning (PHP) algorithms for reducing the generation of candidate itemsets. A new algorithm i.e. Transaction Hashing and Pruning (THP) is also proposed to arrange itemsets in vertical format and use bucket number for hashing transaction id. Zeng et al. [11] proposed the HMT (HASH MAPPING TABLE) and HASH_TREE methodologies are used to optimize the space and time complexity. Rathinsabapathy and Bhaskaran [12] presented MPIP Apriori algorithm. Geng and Tao [13] presented FAHR algorithm based on vector concept with semantic keywords. While assigning the weights to the keywords it reduces the vector dimensions and adds semantic texts to FAHR. Bhandari et al. [14] focus on certain disadvantages of Apriori algorithm and discuss comparative study of six existing improved Apriori algorithms compared with the traditional Apriori algorithm. Deone and Jethan [15] describes the transposition and Boolean matrix technique to get frequent patterns among the larger sets of data in data mining with modification of the Apriori-like algorithm. The proposed method describes the weighted Apriori algorithm based on bit partition technique to reduce the time and space complexity. Due to this the efficiency of the algorithm increases. Qiu-yong et al. [16] proposed the Apriori's optimization algorithm based on reducing transactions. Mundra et al. [17] concentrated more on generating less number of candidate sets and frequent itemsets are calculated in the terms of decimal form, this reduces the number of comparisons and finds the support for frequent itemset.

Sumangali et al. [18] proposed the interesting itemsets algorithm which first performs the preprocessing on database and then remove redundancy among the data. Further the proposed algorithm combines with the FP- tree algorithm for reducing the database scan and produces itemsets from the transactions. The mathematical measures are validated in reduced itemsets. Singh et al. [19] presents the comparison between Apriori and FP-growth algorithm for frequent itemsets generation. Gu Xiao-Feng et al. [20] proposed a new algorithm when the minimum support degree is small; the running speed of the FP-growth algorithm is much faster than the Apriori algorithm. Wang et al. [21] focuses on optimized method which avoids the scanning of database and reduces generation of candidate itemsets repeatedly to improve the performance of algorithm. Here the operational efficiency is still higher than the weighted Apriori algorithm. Singh and Sethi [22] proposed a new approach called Sandwich-Apriori which is a combination of both Apriori and Reverse-Apriori. This Approach reduces number of scans and number of candidates generated as compared to Apriori and Reverse-Apriori. Cheng et al. [23] introduced two different Algorithms based on the concept of higher weight score with the number of frequent itemsets in association rule mining called HWA

(O) and HWA (P) and compared the content of rules between the HWA (O), HWA (P), and Apriori algorithms. Mallik et al. [24] suggested a weighted rule-mining technique to rank the rules in mining algorithms. Weighted rule-mining performs the interestingness measures with weighted support and confidence, these algorithms are mainly used to bypass the problem of rank- based weighted condensed support (WCS) and the weighted condensed confidence (WCC).

III. ANALYSIS OF APRIORI ALGORITHM

Apriori algorithm is used for finding the frequent itemsets among association rules but, this is not the first approach for finding frequent itemsets before that there are two more approaches like, AIS and SETM algorithms. But this algorithms was required more database scans and generate the number of candidate sets. Apriori is mainly works on transactional databases. Each transaction contains a set of items called itemsets. However, the Apriori algorithm takes the transactional datasets and user-defined minimum support as input and produce support, confidence, strong rules and closed itemsets as a result of frequent itemsets. The working of Apriori algorithm is done step by step level-wise for the k itemsets to explore (k+1) itemsets in the database. At every pass of Apriori algorithm after generating the candidate sets it find the new set of frequent items. Firstly in Apriori it scan the database for 1-itemset and also get the counts i.e., support value of each item. This pass is denoted by C1 the first set of candidates. From C1, discover the value of L1 using minimum support value. Generate the value of C2 same as C1 but in this step we have to make pair of items with their count in database. Always preferred the previous pass to generate next one. From C2, generate L2. Likewise, combine items in triples, quadruples and generated L3 this process is repeated till the all combinations of items in transaction are satisfied. At the end of all the passes the set of frequent itemsets are generated. The Apriori algorithm is followed by two important steps like: (1) *Joining Step:* in joining make the possible combinations of itemsets with their respecting support count. the second step (2) *Pruning Step:* Scan the database and check the value of support count is greater than or equal to the user-defined support count value. if 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. Figure 1 shows the example of Apriori algorithm.

Frequent itemsets are very useful to increase the business growth of product. While maintaining the schema of generating information for frequent itemsets is beneficial for business purpose like supermarket, mall, banks, stock markets, etc, To find the frequent itemsets from larger database use the different algorithms of association rule mining like Apriori algorithm, FP-growth algorithm and improved versions of Apriori algorithms etc., Among all these algorithms of association rule mining this paper is mainly focus on basic traditional Apriori algorithm.

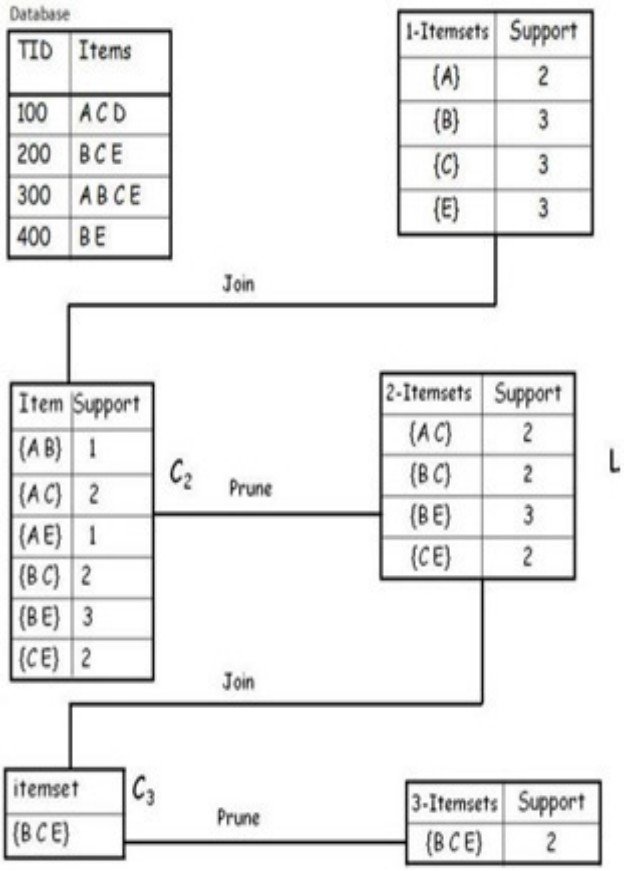

Fig.1. Example of Traditional Apriori Algorithm

The straightforward approach for finding the frequent itemsets is to apply the Apriori algorithm on transactions which is very simple and clear algorithm in data mining. The two important concepts of Apriori algorithm i.e. joining and pruning are used to find the set of frequent itemsets. The experimental results of the Apriori algorithm is given in section 4. But with the ease of Apriori algorithm it also has some limitations.

Limitations of Apriori Algorithm: Apriori algorithm has some limitations as follows: (1) for generating candidate sets, Apriori algorithm requires number of scans over the database. Because of the multiple scanning over database it take lots of time to execute and increases I/O load. (2) Also while scanning databases it generate the number of candidate sets in database.

In order to overcome the drawbacks of Apriori algorithm, there are various types of improvement in this algorithm like matrix, weighted Apriori, hash structure; interest itemsets, transaction compression etc. are possible. But it is found that amongst them, hash structure and weighted Apriori gives better results. On observing above two algorithms it is found that these algorithms are also having some limitations, the benefits of both algorithms could also not bypassed, and not meet the requirement of less computational time and memory

space. Hence, there is a scope to develop new algorithm with the positives of both Weighted Apriori and Hash Tree Apriori algorithms. Individually, they are more efficient, so, to make them more useful, its better to combine both the algorithms to get hybrid algorithm. The main task involved in hybrid algorithm is to address the limitations of normal weighted Apriori and hash-t Apriori algorithms. Limitations associated with these algorithms are as follows. *Weighted Apriori:* (1) Itemset combination will generate frequently. and it will also increase candidate itemsets; (2) The weight of computation for each transaction will take more time to execute; and (3) Not depend on data deviation. *Hash Tree Apriori:* (1) High computational requirement needed; (2) High memory utilization is required; and (3) Node processing requires the high time to compute.

IV. EXPERIMENTAL RESULTS

In Execution of traditional Apriori algorithm the frequent itemsets are generated on retail market datasets. To check the limitations, it is observed that the terms min_support and min_confidence and closed itemsets given by algorithm consumed maximum computational time and memory space. Therefore, traditional Apriori algorithm is not so efficient for larger datasets. Hence, it is found that there is need of better efficient algorithm which will consume less computational time and space to give frequent itemsets. Results of traditional Apriori algorithm are shown in Figure 2 through Figure 5. Generation of itemsets which will be displayed in the list view and on the selection basis the transactions are created which will be added to the database for the further use. Relevant screenshot is shown in Figure 2.

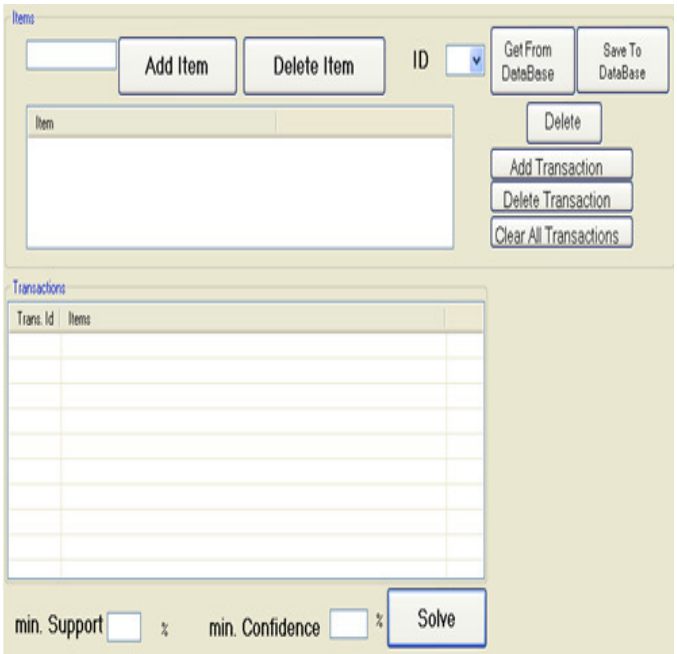

Fig.2. Main Form for Applying Apriori Algorithm

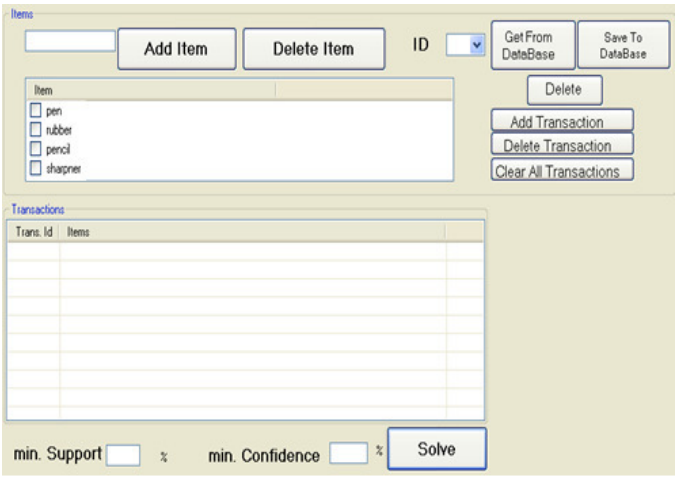

Fig.3. Adding Items for Generating Transactions

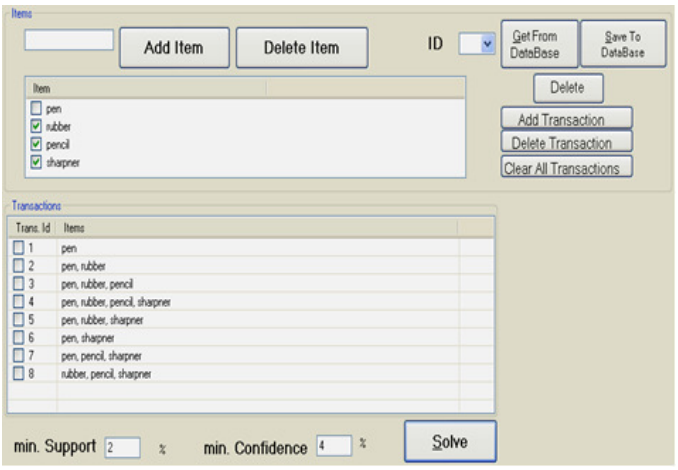

Fig.4. Generating Transactions

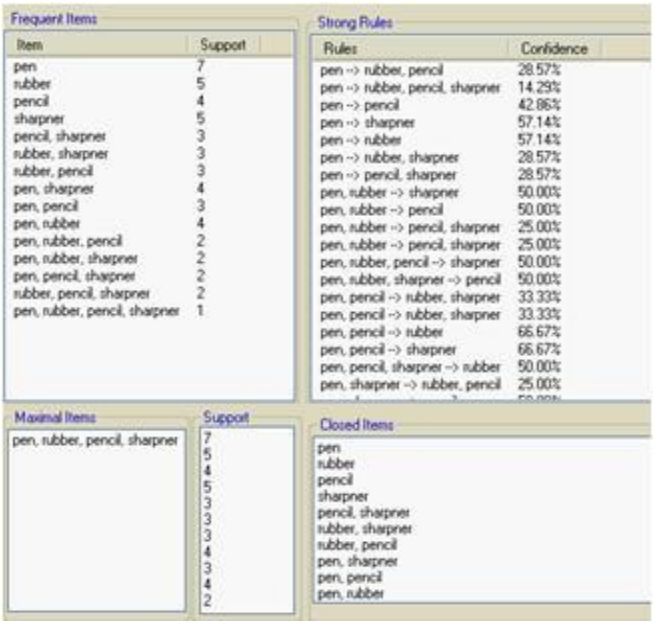

Fig.5. Generating Frequent Itemsets using Apriori Algorithm

Adding items to the item list for generating transaction is shown in Figure 3. For the Apriori application, after adding these transactions to the list, select item as per the individual transaction by add transaction and save it to the database. We can generate transaction or we can get it from the database by using get from database function. Transaction id can be generated or we can use transaction id to get stored transaction from database by providing min_sup and min_conf to the function. Generation of transaction is shown in Figure 4. Calculated support and confidence is shown in Figure 5. Each individual item is calculated by Apriori algorithm and the said transactions can find out the closed itemsets and the maximal itemsets.

V. DISCUSSION

There are many new modifications are possible that can improve the efficiency of Apriori algorithm. The two basics things which are always are the issue in Apriori algorithm i.e. database scanning and generation of candidate sets. So, to deal with them, we can have many improvements in the Apriori algorithm of association rule mining for increasing efficiency of algorithm. But, amongst them, it is found that the weighted value Apriori and hash tree Apriori are the best and they give better efficiency in Apriori. It is observed that individually they are not more efficient as they are together. So, it is better to combine the benefits of both the algorithms to make them more efficient. That is, developing a hybrid approach based on weighted value Apriori and hash tree Apriori algorithm to improve computational time and memory usage. This paper is helpful for the researchers who focus on frequent pattern mining on different transactional datasets i.e. retailing, finance, banking, marketing, insurance, healthcare, etc.

VI. CONCLUSION AND FUTURE WORK

The use of data mining is very wide it helps to growing in business applications, so, the research scope exists for new algorithms applied to the large amount of data stored in enterprise's databases. The main objective of this paper is to observe the various Apriori algorithms and find the limitations of existing Apriori algorithms. From the experimental results, it is observed that the evaluation of support and confidence for traditional Apriori algorithm consumes more time, space and also generates the number of candidate sets.

In future, there is a scope for development of improved or modified version of existing Apriori algorithms based on weighted value Apriori, hash tree Apriori, matrix, interest itemset, transaction compression. One can develop the hybrid algorithm using different existing Apriori algorithms for identifying frequent itemsets with focus on reducing the computational time and memory space using a retail data set and shopping mall dataset.

REFERENCES

[1] Changxin Song, "Research of Association Rule Algorithm Based On Data Mining," *IEEE International Conference of Big Data Analytics (ICBDA)*, Pp.1- 4, 12-14 March 2016.

[2] O. Jamsheela and Raju.G, "Frequent Itemset Mining Algorithms: A Literature Survey," *IEEE International Advance Computing Conference(IACC)*, Pp. 1099-1104, 12-13 June 2015

[3] 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 May 2012.

[4] Archana Singh and Jyoti Agarwal, "Proposed Algorithm for Frequent Item Set Generation," *IEEE International Conference On Contemporary Computing(IC3)*, Pp.160-165, 7-9 August 2014.

[5] S. D. Patil and Dr R. 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, March 2016.

[6] 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.

[7] Yu 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.

[8] Manal Alharbill, Sudipta Pathak and Sanguthevar 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.

[9] 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 August 2015.

[10] 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.

[11] Zhiyong Zeng, Hui Yang, Tao 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.

[12] 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.

[13] Xinqing Geng, Fengmei 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 November 2012.

[14] Pranay Bhandari, K. Rajeswari, Swati Tonge, Mahadev Shindalkar, "Improved Apriori Algorithms – A Survey," *IEEE International Journal of Advanced Computational Engineering and Networking*, Volume-1, Issue- 2, April-2013

[15] Jyoti B. Deone and Vimla 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, March 2014.

[16] H. Qiu-Yong, T. Ai-Long and S. Zi-Guang, "Optimization Algorithm of Association Rule Mining Based On Reducing the Time of Generating Candidate Itemset," *IEEE International Conference On Automation & System Engineering(CASE),* Pp. 1-4, 30-31 July 2011.

[17] 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.

[18] Sumangali. K, Aishwarya.R, Hemavathi.E & 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.

[19] Avadh Kishor Singh, Ajeet Kumar and Ashish K. 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.

[20] Xenia-Feng Gu, Xiao-Juan Hou, Ao-Guang Wang, Hui-Ben Zhang, Xiao-Hua Wu, Xiao-Ming 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.

[21] Pei Wang, Chunhong An and 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.

[22] Tarinder Singh and Manoj Sethi, "Sandwich-Apriori: A Combine Approach of Apriori and Reverse-Apriori," *IEEE India Conference (INDICON),* Pp. 1-4, 17-20 December 2015.

[23] Liewean Cheng, Su-Chuan Chen, And Jashen Chen, "Applying Weighted Association Rules with The Consideration of Product Item Relevancy," *IEEE International Conference On Service Systems and Service Management,* Pp. 888-893, 8-10 June 2009.

Saurav Mallik, Ujjwal Maulik, Anirban Mukhopadhyay, "RANWAR: Rank-Based Weighted Association Rule Mining from Gene Expression and Methylation Data," *IEEE Transactions On Nanobioscience,* Volume 14, Issue 1, Pp. 59-66, January 2015.