Webology (ISSN: 1735-188X)
Volume 18, Number 6, 2021

An Improved Apriori Algorithm For Mining Frequent Pattern

Arpit Solanki and Dr. Rajeev G. Vishwakarma

Department of Computer Science & Engineering, Dr. A. P. J. Abdul Kalam University, Indore (M.P.) - 452010, India.

Corresponding Author: Arpit Solanki

Abstract: In data mining the frequent pattern mining has an essential role in mining transactional database. Among different frequent pattern mining techniques the apriori algorithm has their own importance in mining association rules. The rule miningis used to understand the relationships of dataset variable. In this paper, the issue of running time and memory utilization of apriori algorithm has been addressed. Additionally, an improved apriori algorithm has been introduced for improving the performance of Apriori algorithm. The paper includes a review of the recent research articles based on frequent pattern mining. The aim is to highlight the application, research issues and their possible solution. Next,an improved version of apriori algorithm has been proposed. The experiments have been carried out on UCI based datasets.The experimental results of the improved apriorialgorithms has been measured and compared with three popular frequent pattern mining algorithms. According, to the performance, we found that the proposed apriori algorithm and éclat is efficient algorithm as compared to traditional priori and FP-tree algorithm. Additionally, the modified apriori algorithm has also minimized the problem of information loss.

Keywords: Frequent pattern mining, Association rule mining, Data mining, Rule mining techniques, experimental comparison.

I. INTRODUCTION

The data mining is used for data analysis and pattern recovery. The algorithms applied on the data to explore and recover the essential insights. There are a number of algorithms are available for the data analysis task. The selection of an algorithm for data analysis has been done according to the nature of data analytics application such as classification, clustering, prediction and rule mining. The rule mining techniques are mainly used understand the relationship of data variables. In this paper, an association rule mining techniques is the key area of interest. The association rule mining which is also abbreviated as frequent pattern mining is the process of identifying patterns within a dataset that occur frequently. This is done by finding items that appear together.This technique is useful for discovering the information, which is hidden. These techniques have a large range of applications such as**Market Basket Analysis to**analyze customer purchasing patterns, **Recommender Systems** tosuggest appropriate product to consumer,

Network Intrusion Detectionto identify a security threat, **Medical Analysis** to identify patterns to indicate a particular disease, **Text Mining**to identify patterns in text and **Web usage mining** to analyze patterns of user behavior on a website.

But there are some limitations in such pattern mining task such as:a large number of patterns with high dimensions, large data processing time, and huge space complexity. Additionally, the number of patterns can be very large, making it difficult to interpret the results. In this paper, to address these issues in association pattern mining we considered the apriori algorithm.Apriori algorithm is easy way to mine frequent itemsets. The algorithm performs searches in database to find frequent itemsets. Each itemset must be greater than or equal to a minimum support threshold. First, the algorithm scan database to find frequency of 1-itemsets (only one item) by counting in database. The frequency of 1-itemsets is used to find the itemsets in 2- itemsets which in turn is used to find 3-itemsets and so on. The generation and evaluation of the itemset is a time consuming process as well as when the thresholds are applied on mined itemsets the information loss has become a key issue for achieving accurate decision making. Therefore, the proposed work is aimed to design and implement a modified apriori algorithm for efficient and effective frequent pattern mining. This section discusses the basics of the frequent pattern mining and the applications. The next section includes the related study which is recently contributed for improving the frequent pattern analysis algorithm.

II. RELATED STUDY

This section offers the background study related to the frequent pattern mining and association rule mining.

A. Essential Keywords

Table 1 List of keywords

Abbreviation	Full form
ARM	Association Rule Mining
DNA	Deoxyribonucleic Acid
FPM	Frequent Patterns Mining
FSG	Frequent Sub Graph Discovery
FU-tree	Frequency-Utility Tree
GPM	Graph Pattern Mining
HUOPM	High Utility Occupancy Pattern Mining
NetNCSP	Nettreefor Non-Overlapping Closed Sequential Pattern
PDR	Peak Demand Reduction
PSO	Particle Swarm Optimization
PSPM	Parallel Sequential Pattern Mining
SPP	Safe Pattern Pruning

SPM	Sequential Pattern Mining
TOU	Time Of Use

B. Recent Review

S. Kumar et al [1]give an overview of the distinct approaches to pattern mining in Big Data. They investigate the problem of pattern mining and associatedtechniques. Then, examine developments in parallel, distributed, and scalable pattern mining, in the big dataperspective. They study four varietiesof itemsets mining. They conclude with openissues and opportunity. It also provides direction for enhancement.

F. Min et al [2] proposes a frequent pattern discovery algorithm by dividing the alphabet into strong, medium, and weak parts. Experiments were undertaken on data in various fields to reveal the pattern. These include protein sequence, petroleum production, and Chinese text. The results show tri-patterns are meaningful andenriches the semantics of applications.

The frequent itemsetsis time consuming with increase data and memory is needed in mining due to computation. Therefore, an efficient algorithm is required to mine frequent itemsets in a shorter run time and less memory. **C. H. Chee et al [3]** compare different algorithms for Frequent Pattern Mining so that a more efficient algorithm can be developed.

SPM neglects repetition in sequence. To solve this problem, gap constraint SPM was proposed by **Y. Wu et al [4]** and avoids finding useless patterns. Author adopts closed pattern mining and proposes an algorithm NetNCSP. Ithas two steps, support and closeness calculation. Backtracking is used to calculate non-overlapping support, which reduces time. They also propose pruning, inheriting, predicting, and determining. The pruning is able to find the redundant patterns and can predict the frequency and closeness. Results show that it is efficient and discover closed patterns.

V. Dias et al [5] propose high performance and productivity system for distributed GPM. It employs a dynamic load-balancing based on a hierarchical and locality-aware work stealing mechanism, to adapt workload characteristics. Additionally, it enumerates sub-graphs by combining a depth-first strategy with scratch processing to avoid storing large amounts of intermediate state andimproves memory efficiency. For programmer, it presents an intuitive, expressive and modular API. The implementations outperform on many problems to sub-graph querying.

To extract high quality patterns, **W. Gan et al [6]** extends the occupancy measure to assess the utility of patterns. They propose an algorithm HUOPM. It considers user preferences like frequency, utility, and occupancy. A FU-tree and two data structures, called the utility occupancy list and FU-table, are designed for pruning. It can discover the complete set of high quality patterns without candidate generation. Results show that the patterns are intelligible, reasonable and acceptable, and with its pruning it outperforms, in terms of runtime and search space.

FPM and ARM have problems like memory cost, processing speed, and space. **W. Gan et al [7]**, survey current status of PSPM, including categorization and parallel SPM. They review parallel SPM, including partition, Apriori, pattern growth, and hybrid algorithms, and provide advantages, disadvantages and

summarization. Topics, including parallel quantitative / weighted / utility SPM, uncertain data and stream data, hardware acceleration, are reviewed.

R. Bunker et al [8] apply sequential pattern mining algorithm called SPP to 490 labeled event representing passages of rugby team's matches. They obtain patterns that are the discriminative between scoring and non-scoring outcomes and opposition teams' perspectives, and compare with frequent patterns. From results, successful line-outs, regained kicks in play, repeated phase-breakdown, and failed plays. Because of the pruning, compared to the patterns of unsupervised methods, were more sophisticated and useful.

T. A. D. Lael et al [9] analyze consumer purchasing patterns for motorcycle parts using FP-Growth algorithm. The aim is to obtain information for marketing and sales. The results show that the FP-Growth can be used to identify consumer purchasing patterns. Some patterns found a combination of often purchased products, purchase time, and category. FP-Growth can assist in understanding consumer purchasing patterns to improve the marketing and sales.

C. R. Wijesinghe et al [10] identify the frequent workflow patterns in a corpus of Galaxy bioinformatics workflows. FSG algorithm is used. Seventy-one reusable workflow patterns identified with a 5% minimum support. Future plan is to annotate the identified patterns and encode in the workflow with objective of improving the usability to user.

A. Yang et al [11] introduces process of sequencing technology, for DNA sequence data structure and similarity. They analyze data mining, ML, and put the challenges in biological datamining. Then, review 4 applications of ML in DNA data: sequence alignment, classification, clustering, and pattern mining. They analyze biological application and significance, and summarized development and problems.

F. Wang et al [12], customer PDR characterizing model is proposed, where difference-in-difference model is adopted to quantify the effect and probability fitting method is used to characterize the feature. AnARM analysis using Apriori algorithm to explore the impacts: dwelling characteristics, socio-demographic, appliances and heating, and attitudes towards energy. Results based on 2993 records containing smart metering data that PDR level cannot be obtained based on ownership and usage. The framework improves benefits of TOU programs and guide policy makers.

Y. Ali et al [13] investigates the recovery and death factors to schistosomiasis disease dataset, collected from Hubei, China. Apriori is used to spot factors. Different tools were used for analysis and evaluation with minimum support and minimum confidence indicated higher than 90% to generate rules. In addition, attributes indicating recovery and death. Results may useful for in precise treatment decision.

M. H. Santoso et al [14]performed data mining to determine the minimum valueof support and confidence that will produce theassociation rule. The rule is used to produce thepercentage of purchasing activity for an itemset within aperiod of time using RapidMiner. By searching for patterns using apriori algorithm, the resulting information canimprove further sales strategies.

According to **S. Das et al [15]** in 2011, 4,432 pedestrians were killed, and 69,000 were injured. Alcohol was involved in nearly 44%. Authors utilized 'Apriori' to discover crashpatterns by using 8 years of data.The results indicated that road lighting at night helped inreducing crash. Male greatersusceptibility

towards severe and fatal crashes, younger female more crash-prone,vulnerable impaired even on road lighting, middle-aged male being inclinedtowards crash, and dominance of single vehiclecrashes. Findings will help professionals in understanding patterns andsolution.

According to **Y. Zhou et al [16]**traditional Aprioriis slowdue to scanning and excessive candidate item-sets.A book management system based on improved Apriori algorithm is designed. The information of the borrowers and booksextracted from books lending database. The data is cleaned, converted and integrated, to mine the rules.The association matching is performed according to the borrower and selected books. The book associated with the borrower is recommended. The results show that the system can recommend book, and CPU occupationis only 6.47%.

O. S. Adebayo et al [17] to improve the detection rateof malicious application, knowledge-based database discovery model that improves aprioriassociation rule mining with PSO is proposed. PSO is used to optimize the candidate and parameters ofapriori algorithmfor features selection. The candidate detectors generated by PSO form apriorirule.These rules are used with extraction algorithm to classify and detect malicious android application.The results show that the proposed apriori rule with PSO model has remarkable improvement.

The complexity and execution time are the majorfactors in data mining. The results demonstrate that the precision ratio of the presented technique is highcomparable to other existing techniques with the same recall rate, i.e., the R-tree algorithm. **Z. Zhao et al [18]**proposed a technique by which the mining algorithm controls the noisy data, and precision is also very high. Author makes a systematic and detailed analysisof data mining technology by the Apriori algorithm.

To identify the key causes to accidents, it is necessary to minethe association rules between risk factors of theaccidents. **Q. Ca et al [19]** improves the Apriori algorithm to mine theassociation rules, and probes the causes of traffic accidents.According to the layer and dimension of attributes, the parameters like support, confidence and liftwere adjusted. The results were screened toobtain a series of association rules. The results enable the traffic department to formulateaccident control measures, and traffic safety.

Y. Le [20], data mining with Hadoopis applied to data resource management platform of biomass energy, which solves mass data collection, storage, processing and analysis. The Apriori algorithmin big data lies in high time complexity. The author combines Apriorialgorithm with Hadoop, and test the algorithm. The test realized the design of storage mode of database, and performance was improved by 40% compared with traditional Apriori algorithm.

J. Yang et al [21], aiming at time and poor monitoringaccuracy in customer service data.A analysis platform operation abnormalitymonitoring method based on improved FP-Growth is designed. Obtain data sets, classify data types, filter customer behavior, identify the operating status, improve the FP-Growth, set the platform safety, and keep low error, optimize the monitoring mode.

How to efficiently analyze the frequency of drug use, combination, and association between drugs is a core problem in prescriptioncompatibility. **S. Zhou et al [22]**, study compatibility rules of Chinese antiviralprescriptions and mechanism of medicine molecules. FP-growth was used to analyze 961 prescriptions. First, FP tree was constructed then, rules were established. Finally,frequency and association

rules were analyzed according to dosage. The results show that the FP-growth algorithm has excellent performance and stronggeneralization and robustness.

III.PROPOSED WORK

Before discussion of the proposed modification on apriori algorithm, let we discuss the working of apriori algorithm for frequent pattern generation. Apriori assumes that all subsets of a frequent itemset must be frequent.If an itemset is infrequent, all its supersets will be infrequent.Consider the following dataset and using this dataset we are trying to generate association rules:

Table 2 Example Dataset

Transaction Id	Items
T_1	A_1, A_2, A_5
T_2	A_2, A_4
T_3	A_2, A_3
T_4	A_1, A_2, A_4
T_5	A_1, A_3
T_6	A_2, A_3
T_7	A_1, A_3
T_8	A_1, A_2, A_3, A_5
T_9	A_1, A_2, A_3

The rule generation using the dataset and apriori algorithm we need two additional parameters support and confidence. Let support count is 2 and confidence is 60%. First, we generate a table, which contain count of each item belongs to the dataset. This table is known as candidate set. The first candidate set (C_1) is given in table 3.

Table 3 candidate set (C_1)

Item set	Frequency (support count)
A_1	6
A_2	7
A_3	6
A_4	2
A_5	2

Next, we compare candidate set item's support count with minimum support count. Here, we considered $\min_{support} = 2$.If candidate set item's support count is less than $\min_{support}$ then we remove items. This gives us itemset L_1. In candidate set C_1 has satisfy the condition of support threshold $\min_{support}$. In next step, we needed to generate candidate set C_2 using L_1.This process is kwon as join. Before. Joining we need to check the condition, where L_{k-1} should have $(K-2)$ elements in common.

Thus, we need to check all subsets of an itemset are frequent or not.If not frequent then we remove that itemset.Now find support count of these itemsets. The table 4 contains the candidate set C_2.

Table 4 candidate set C_2

Item set	Support count
$\{A_1, A_2\}$	4
$\{A_1, A_3\}$	4
$\{A_1, A_4\}$	1
$\{A_1, A_5\}$	2
$\{A_2, A_3\}$	4
$\{A_2, A_4\}$	2
$\{A_2, A_5\}$	2
$\{A_3, A_4\}$	0
$\{A_3, A_5\}$	1
$\{A_4, A_5\}$	0

Next we compare candidate (C_2) item set's support count with minimum support.It candidate set item is less than $\min_{\text{support}}$ then we remove those items.This process generate the itemset L_2.

Table 5 next generation itemsets(L_2)

Item set	Support count
$\{A_1, A_2\}$	4
$\{A_1, A_3\}$	4
$\{A_1, A_5\}$	2
$\{A_2, A_3\}$	4
$\{A_2, A_4\}$	2
$\{A_2, A_5\}$	2

Next, algorithm generates new candidate set C_3using L_2based itemsets. Check if all subsets of these itemsets are frequent or not and if not, then remove that itemset.

Table 6 candidate set C_3

Item set	Support count
$\{A_1, A_2, A_3\}$	2
$\{A_1, A_2, A_5\}$	2

The candidate set (C_3) is now compared with the minimum support count, which results similar set of itemsets as C_3, thus$C_3 = L_3$. Next we need to create the next candidate set C_4, but we found the frequency of the combination of itemset are less than the support threshold. Therefore,C_4 is not generated. Finally,

these generated frequent patterns are used to provide decision rules these rules. In order to identify the strong rules the confidence has been used. The confidence is considered here as 60%. Additionally the confidence of a rule is calculated using the following formula:

$$\text{conf}(\text{A} \rightarrow \text{B}) = \frac{\text{Sup}(\text{A} \cup \text{B})}{\text{Sup}(\text{A})}$$

Finally, we take an example to create association rules we take the itemset from table 6 $\{A_1, A_2, A_3\}$. Using this we can generate the following set of rules:

$$A_1 \& A_2 \rightarrow A_3$$

$$A_1 \& A_3 \rightarrow A_2$$

$$A_2 \& A_3 \rightarrow A_1$$

$$A_1 \rightarrow A_2 \& A_3$$

$$A_2 \rightarrow A_1 \& A_3$$

$$A_3 \rightarrow A_1 \& A_2$$

Here, using the itemset we can generated 6 rules thus in order to prune the rules the confidence of the rules has been calculated. In our example the confidence of rules is: 50%, 50%, 50%, 33%, 33%, and 33%. Now by using the confidence threshold we prune the amount of rules which are less effective. But this step can cause the information loss and sometimes the false alarm rate can increases. Therefore, in this presented work, we proposed a modified apriori algorithm for minimizing the information loss and improving the accuracy of algorithm. The steps of the proposed modified apriori algorithm are given below:

Table 7 Proposed Apriori Algorithm

Input: Dataset D, Support Threshold T, confidence threshold conf **Output:** SList, FList
Process: 1. Find total symbols (S) in dataset (D) 2. Apply support threshold to filter symbols 3. While flag $==$ true a. if condidate $=$ true a. Generate candidate set C b. Filter C using support threshold L c. F = F + L b. Else

a. Flag = false

c. End if

d. While next

4. End while
5. Use F to generate association rules R_n
6. For$(i = 1, i \leq n; I++)$
   a. $conf = CalculateConf(R_i)$
   b. if $conf <$ confidence threshold
      a. $SList.Add(R_i)$
   c. else
      a. $FList.Add(R_i)$
   d. end if
7. end for
8. Return SList, FList

The proposed algorithm returns two list as compared to one list of rule. The list SList contains the rules those have confidence less then support confidence, additionally the high quality rules are kept on the FList. During the utilization of these rules first the FList rules are being used and if the data has not satisfied than the SList rules are used. That process enhances the reorganization process and improves the accuracy of rule based classification

Next this algorithm has implemented using python technology and for comparison the proposed algorithm, it is compared with traditional Apriori, FP-tree and Eclat algorithm. the brief introduction of algorithms has given below:

FP-Tree algorithm: Frequent Pattern Tree is a tree-like structure that is made with the initial itemsets of the database. The purpose of the FP tree is to mine the most frequent pattern. Each node of the FP tree represents an item of the itemset. The root node represents null while the lower nodes represent the itemsets. The association of the nodes with the lower nodes that is the itemsets with the other itemsets is maintained while forming the tree. The frequent pattern growth method lets us find the frequent pattern without candidate generation.

Eclat Algorithm: The ECLAT algorithm stands for Equivalence Class Clustering and bottom-up Lattice Traversal. It is a popular method of Association Rule mining. It is a more efficient and scalable version of the Apriori algorithm. The Apriori algorithm works in a horizontal sense imitating the Breadth-First Search, the ECLAT algorithm works in a vertical manner like the Depth-First Search. This approach of the algorithm makes it a faster than the Apriori algorithm.

IV. RESULTS ANALYSIS

In this section, we investigate the performance of the proposed association rule mining algorithm. Additionally a comparison with the traditional association rule mining algorithms has also been presented. The implemented algorithms have been also introduced in previous section. In this section the experimental results has been discussed.

A. Experimental scenarios

The publically available frequent pattern datasets are utilized from UCI repository. Additionally the performance in terms of number of rules generated and time utilization to generate the rules has been evaluated. There are three experimental scenarios have been considered:

1. **Scenario A:**The number of items is increasing for performing variations on the training data. During this process the performance in terms of number of rules and training time is recorded.
2. **Scenario B:**The support threshold has been increased and the number of rules and training time has been recorded.
3. **Scenario C:**Confidence threshold variation and performance evaluation has been done to observe the influence of confidence in number of rule generation and training time.

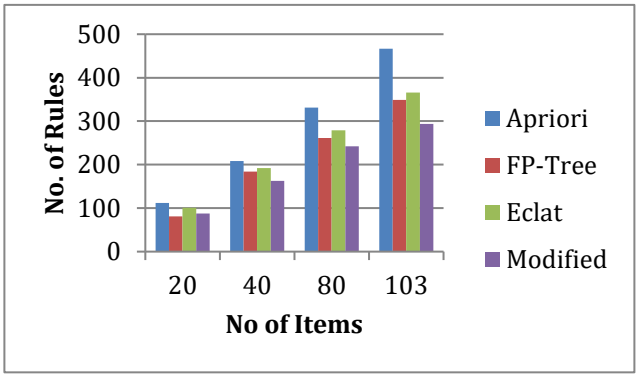
B. Performance parameters

In order to describe the performance of the proposed algorithm two key parameters has been considered i.e. training time and number of generated rules. The number of rules counting is help to understand the influence of the thresholds in data loss. Additionally,the training time hasshows computational cost of the algorithm. It is the amount of time to mine the rules. The time can be measured using:

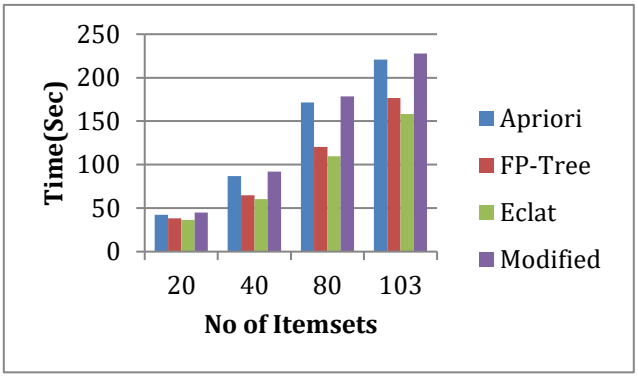
$$\text{time consumed} = \text{End time} - \text{Start Time}$$

C. Results discussion

In this paper three set of experiments has been performed to evaluate the performance of the proposed algorithm. Theexperimental performance of the experiments based on the scenarios has been discussed as:



(A)



(B)

Figure 2 Number of generated frequent pattern rules by variations of (A) Number of itemsets (B) Filtering Threshold

Scenario A:

In this experiment the aim is to increase the number of unique symbols in dataset. Additionally observe the influence on the performance. Figure 2 demonstrate the performance of the comparative performance in terms of generated number rules and the time to generate rules. Therefore, the number of unique dataset symbols (itemsets) is varied, and with the variable size of dataset experiments has been performed. Figure 2(A) shows the number of rule generated. The X axis shows the number of unique symbols used in dataset and Y axis shows the number of rules generated. According to the obtained performance the proposed method generate fewer amounts of rules and we can use high confidence threshold for fast data analysis. The model provide the ability to prepare high quality rules for faster processing at the same time when the misclassification happen it can use the secondary list of rules for classification.

On the other hand the time requirement of rule generation algorithms has been compared in figure 2(B). The X axis contains the number of unique symbols in dataset and Y axis shows the training time of the rule generation algorithms. According to the bar graph representation off the time consumption we can see the proposed algorithm is consuming higher amount of time for generating the rules. But is slightly high as compared to apriori algorithm but the detection performance of the model has improved.

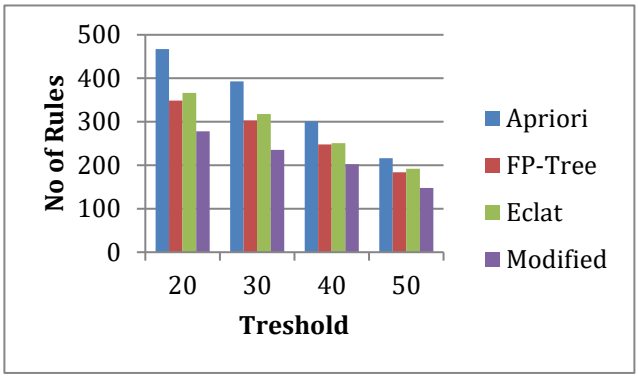
Scenario B:

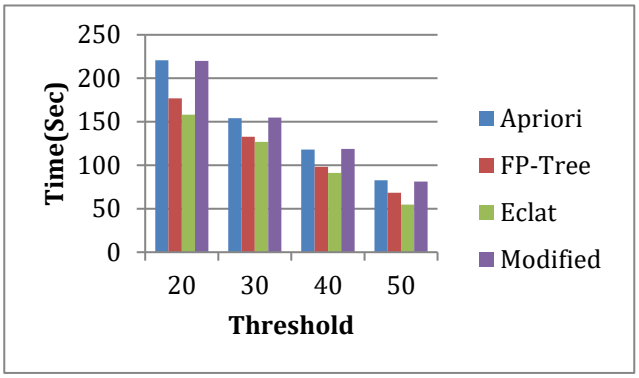
In this scenario, performance influence by varying the support threshold has been discussed. The performance of the algorithms in second scenario is given in figure 3. In this figurenumber of rule generated by increasing support thresholdhas given in Figure 3(A). The support threshold is given in terms of percentage (%). The X axis shows the support threshold and Y axis shows the number of rule generated. According to the results, we found that the number of rules is increasing with the amount of items involved. Additionally, when the support threshold has been increased then the number of generated rules is reducing in the similar ratio.

The time consumed is given in figure 3(B) and measured in terms of seconds (sec). Figure 3(B) shows the time consumed by different rule mining algorithms. Figure 3(B) shows the training time by varying the support threshold. In this diagram the Y axis contains the time taken and X axis shows the support threshold. According to the results the training time is increasing with the amount itemsets and reducing with the increasing thresholds. According to the obtained results the proposed algorithm consumes similar amount of time as the apriori algorithm for generating the rules.

Scenario C:

In this scenario of experiment the algorithm is evaluated for increasing number of confidence value for rule generation. the figure 4 shows the performance of the algorithms in terms of number of rule generated and required training time. The figure 4(A) shows the number of rules generated with increasing number of confidence. The confidence has considered here in terms of percentage (%).

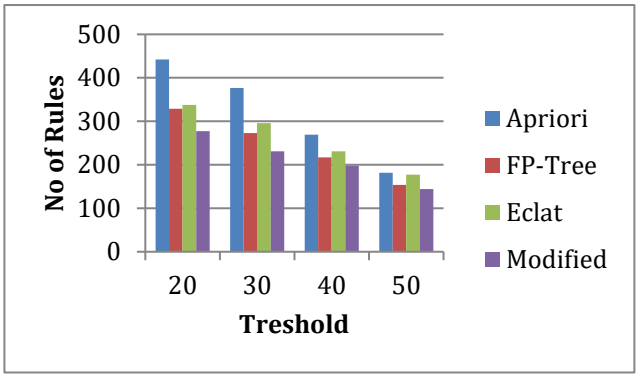


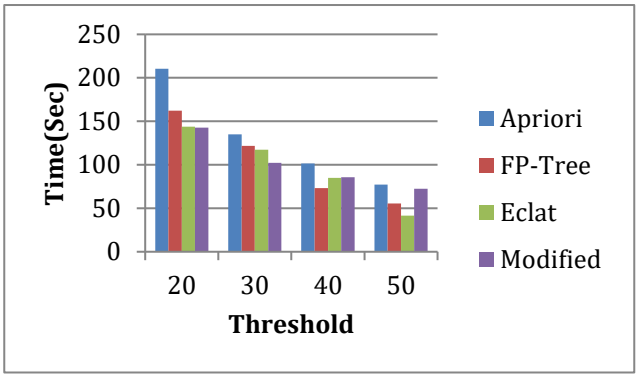


(A) (B)

Figure 3 shows the training time for frequent rule mining techniques by variations of (A) Number of itemsets (B) Filtering Threshold

The next figure 4(B) includes the training time of the algorithms when the number of confidence threshold has been increased. According to the experimental results, the proposed algorithm shows the confidence value will significantly change the number of rules generated.





(A) (B)

Figure 4 shows the performance of rule mining techniques in terms of (A) Number of rules (B) Confidence Threshold

According to the obtained results the proposed technique is producing much consistent amount of rules as compared to the other implemented algorithm. The other algorithms are frequently changing the number of rules with the increasing and decreasing number of confidence. But the proposed technique is not fluctuating much and produces the consistent results. In addition, when we discuss the performance in terms of training time the training time with the increasing confidence has reduces significantly.

V. CONCLUSION & FUTURE WORK

The frequent pattern mining or association rule mining technique has a wide area of applications. This technique is used to establishing relationship among items. These rules can be used in various applications for frequent decision mining. In order to mine rules from a transactional dataset a number of algorithms are available. But most of the rule mining technique has suffers from the data loss, low accuracy and high running time. Therefore this paper introduced a modified apriori algorithm. the paper is organized in three main parts. First part includes a review of different rule mining techniques. Therefore, review of recent applications of association rule mining and frequent pattern mining algorithms in real worlds has been conducted. Next a detailed overview of apriori algorithm has been discussed and the example of apriori algorithm has used to explain the details of apriori algorithm. Next the apriori algorithm has modified and the algorithm steps of the modified apriori algorithm have been given.

After implementation of the modified apriori algorithm a comparative performance analysis has been done in comparison with Apriori, FP-Tree and Eclat algorithms.Additionally, to conduct experiments the dataset available in UCI repository has been used. Based on the experimental results we summarize our findings as:

1. The increasing number of itemsets (unique symbol)are highly influencing the performance in terms of number of rules and training time. The increasing number of unique symbols will increase the required training time and also increases the number of rules generated.
2. The increasing support threshold values can reduce the amount of time consumption and number of rules generated.
3. The increasing confidence threshold also reduces the number of rule generated and training time.
4. The proposed technique is generating a reliable set of rules and not varying much in terms of numbers.

In this paper due to limitations of traditional apriori algorithm a modified apriori algorithm has been proposed. the proposed algorithm is promising, efficient and producing reliable and consistent consequences. Therefore, the proposed algorithm has applied on a real world problem for demonstrating the effectiveness of the proposed algorithm.

REFERENCES

[1] S. Kumar, K. K. Mohbey, “A review on big data based parallel and distributed approaches of pattern mining”, ResearchGate, 34 (6) 2019.

[2] F. Min, Z. H. Zhang, W. J. Zhai, R. P. Shen, “Frequent pattern discovery with tri-partition alphabets”, Information Sciences 000 (2018) 1–18

[3] C. H. Chee, J. Jaafar, I. A. Aziz, M. H. Hasan, W. Yeoh, “Algorithms for frequent itemset mining: a literature review”, ArtifIntell Rev (2019) 52:2603–2621, https://doi.org/10.1007/s10462-018-9629-z

[4] Y. Wu, C. Zhu, Y. Li, L. Guo, X. Wu, “NetNCSP: Nonoverlapping closed sequential pattern mining”, Knowledge-Based Systems 196 (2020) 105812

[5] V. Dias, C. H. C. Teixeira, D. Guedes, W. Meira Jr., S. Parthasarathy, “Fractal: A General-Purpose Graph Patern Mining System”, SIGMOD ’19, June 30–July 5, 2019, Amsterdam, Netherlands, ACM

[6] W. Gan, J. C. W. Lin, P. F. Viger, H. C. Chao, P. S. Yu, "HUOPM: High Utility Occupancy Pattern Mining", Journal of Latex Class Files, VOL. 14, NO. 8, AUGUST 2015

[7] W. Gan, J. C. W. Lin, P. F. Viger, H. C. Chao, P. S. Yu, "A Survey of Parallel Sequential Pattern Mining", ACM Trans. Knowl. Discov. Data., Vol. 0, No. 1, Article 00. Publication date: August 2018.

[8] R. Bunker, K. Fujii, H. Hanada, I. Takeuchi, "Supervised sequential pattern mining of event sequences in sport to identify important patterns of play: An application to rugby union", PLoS ONE 16(9): e0256329

[9] K. Selva Kumar, "Consumer Behvior Analysis using Big Data Analytic", IAIC Transactions on Sustainable Digital Innovation (International Journal of Control Theory and Applications, Vol. 10 No. 30, 2017

[10] C. R. Wijesinghe, A. R. Weerasinghe, "Mining Frequent Patterns in Bioinformatics Workflows", International Journal of Bioscience, Biochemistry and Bioinformatics, Volume 10, Number 4, October 2020

[11] A. Yang, W. Zhang, J. Wang, K. Yang, Y. Han, L. Zhang, "Review on the Application of Machine Learning Algorithms in the Sequence Data Mining of DNA", Front. Bioeng. Biotechnol. 8:1032, 2020

[12] F. Wang, L. Liu, K. Li, N. Duić, M. S. khah, J. P. S. Catalão, "Impact Factors Analysis on the Probability Characterized Effects of Time of Use Demand Response Tariffs Using Association Rule Mining Method", Energy Conversion and Management 197, 2019

[13] Y. Ali, A. Farooq, T. M. Alam, M. S. Farooq, M. J. Awan, T. I. Baig, "Detection of Schistosomiasis Factors Using Association Rule Mining", IEEE access VOLUME 7, 2019

[14] M. H. Santoso, "Application of Association Rule Method Using Apriori Algorithm to Find Sales Patterns Case Study of Indomaret Tanjung Anom", brilliance research of artificial intelligence, Volume 1, Number 2, November 2021

[15] S. Das, A. Dutta, R. Avelar, K. Dixon, X. Sun, M. Jalayer, "Supervised association rules mining on pedestrian crashes in urban areas: identifying patterns for appropriate countermeasures", International Journal of Urban Sciences, 2018

[16] Y. Zhou, "Design and Implementation of Book Recommendation Management System Based on Improved Apriori Algorithm", Intelligent Information Management, 2020, 12, 75-87

[17] O. S. Adebayo, N. A. Aziz, "Improved Malware Detection Model with Apriori Association Rule and Particle Swarm Optimization", Hindawi Security and Communication Networks Volume 2019, Article ID 2850932, 13 pages

[18] Z. Zhao, Z. Jian, G. S. Gaba, R. Alroobaea, M. Masud, S. Rubaiee, "An improved association rule mining algorithm for large data", Journal of Intelligent Systems 2021; 30: 750–762

[19] Q. Ca, "Cause Analysis of Traffic Accidents on Urban Roads Based on an Improved Association Rule Mining Algorithm", Special Section on Big Data Technology and Applications in Intelligent Transportation, Vol. 8, 2020

[20] D. Yu, "An overview of biomass energy research with bibliometric indicators", Energy & Environment, 29, 4 (2018).

[21] G. P. Wang “An Anomaly Detection Framework for Detecting Anomalous Virtual Machines under Cloud Computing Environment”, International Journal of Security and Its Applications, 10, 1, (2016) 75.

[22] S. Zhou, “Data Mining and Analysis of the Compatibility Law of Traditional Chinese Medicines Based on FP-Growth Algorithm”, Hindawi Journal of Mathematics Volume 2021, Article ID 1045152, 10 pages