IJCTA, 9(27), 2016, pp. 513-521 ISSN: 0974-5572

Association Rule Generation Using E-ACO Algorithm

M. Sathya[1] and K. Thangadurai[2]

ABSTRACT:

Over recent years the popularity of frequent itemset mining has soared. In particular many frequent itemset mining and methods for optimization of association rules using Apriori and Eclathave been introduced. In this paper, we propose an integrated Eclat and ACO technique based on an optimization pattern. In contrast to well-known measures from the literature, our technique considers the integration of Eclat and ACO (E-ACO) for optimized rule generation rather than just mining frequent itemsets.We conducted a set of experiments, testing effectiveness on Belgian retail supermarket storedatasets. Based on our experiments results, we obtained good performance in terms of memory and running time. Andoptimizing the number of rules generated.

Keywords: Data mining, Apriori, Eclat, Association rule generation

1. INTRODUCTION

In this paper we study the discovery of frequent items and based on it optimized rules are generated. Many algorithms have been proposed for mining frequent items, such as Frequent Item Sets using Equivalence Pruning (FIS-EP) [1], Frequent Item Sets using Node sets (FIS-N) [2]. By applying N-list, FIS-EP reduced the search space time through Child Parent Equivalence Pruning. On the other hand, FIS-N using pre order and post order code showed high performance on both running time and memory usage. However, both the above methods suffered from the optimized rule generation.

Given the training and testing dataset, the aim of data mining is to discover frequent itemsets aiming at optimizing the rule generation from a huge pile of data. In this aspect, for the frequent itemset domain many studies have been conducted which aim at optimizing the rule generation. Ant colony optimization for travelling salesman problem [3].Under random and cyclic dynamic environments with pheromone trails based on the immigration scheme was applied. Contrary to this in [4], divide and rule for travelling salesman problem using ACO was applied resulting in the running speed. To address memory issues, hybrid monkey search [5] algorithm was applied based on artificial bee resulting in better runtime also.

The ability to discover the frequent itemset and association rule has implications beyond setting the best parameters for data mining algorithms. For instance, it can help characterize the customer behavior pattern in a manner that is useful in increasing the sales boost. Frequent itemset mining for market basked data [6] when applied to gene expression data analysis resulted in efficient pattern detection in a fast manner. Frequent Pattern Growth [7] algorithm for association rule mining resulted in the minimization of execution time by scanning the database only once.

In [8], survey of data mining algorithms for mining medical records was analyzed. However, the question of obtaining the optimized rule for a given dataset still remains unaddressed. Compared to the association rules in

1 Ph.D Research Scholar, PG and Research Department of Computer Science, Government Arts College (Autonomous), Karur, India, Email ID: sathyabala333@gmail.com.

2 PG and Research Department of Computer Science, AssistantProfessor & Head, Government Arts College, Karur, Email ID: ktramprasad04@gmail.com.

horizontal database only few works [9], focus on the association rule mining for optimized message size. Furthermore and as highlighted in [10], sequences of frequent itemset are helpful in real world applications like Field Programmable Gate Array (FPGA), Location Based Service Environments (LBSE) [11].

The remaining part of the paper is organized as follows: Section 2 gives a literature overview. Section 3describes an integrated technique, Eclat and ACO (E-ACO) for optimized rule generation in the context of frequent itemset mining. Section 4 demonstrates the performance of E-ACO technique and comparison made with the state-of-the-art methods. Section 5 then offers an explanation for this performance through an analysis with the aid of graph. Finally, Section 6concludes the paper.

2. RELATED WORKS

Several sequential pattern mining approaches are regarded in the literature, Node linkage approach [12] for sequential pattern mining used Depth First Search algorithms to reduce search space and therefore frequent items. Dynamic load balancing for association rule mining [13] further reduced the execution time for rule generation using a real grid. Mining of relational patterns from multi relational database [14] used tree-based data structure for extracting frequent relational patterns. As already mentioned these methods perform sequential mining and therefore extract the rules.

As discussed before, its cause lies primarily in the redundancy in the returned itemsets. This has long since been recognized as a problem and has received ample attention.To address this problem, graph mining in configuration management databases [15] and cognitive process [16] for identifying customer choice and multiple Graphics Processing Units (GPU) [17] have been proposed which provide maximum parallelism resulting in extracting the representative frequent patterns. However, the frequent items by these methods deteriorate with the frequent scans. Along these lines, Dynamic Bit Vector (DBV) [18] approach and max frequency measure [19] provided an insight into memory and execution time with which the rules were fetched. In [20], fuzzy and granular approach was presented to improve the reliability of the items being extracted.

Customers are segmented and association rules are separately generated to satisfy their specific needs in a cost effective manner using some special promotions for the common groups. From the results it is shown that the market basket analysis using Eclat algorithm for supermarket improves its overall profits [21].

3. PROBLEM STATEMENT

One of the most important tasks considered in data mining is the efficient mining of association rules. A simple association rule can be represented as: '*Sony* $\rightarrow$ *Power Bank*' [support=0.1, confidence=0.8]. The above rule states that there exist a strong association between buying patterns of Sony and Power Bank. With the support indicating that Sony and Power Bank come together in 10% of the transactions and the confidence that Sony has taken part in transactions where Power Bank is also present, 80% of the trans-actions included Power Bank with Sony.

With this rule, we can assume that in the future, those who purchase Sony are most likely to purchase Power Bank. Such information help the retailers explore opportunities for cross-sellingand therefore improve the sale growth from the angle of sales in super market. Let 'I' represent the set of items. A dataset 'D' over '$I = i_1, i_2, ..., i_n$' represents the itemset and '$T = t_1, t_2, ..., t_n$' symbolizes the set of transactions which contains items from 'I'. Then a transaction 'T' over 'I' is a couple '(tid, I)' where 'tid' represents the unique transaction identifier whereas 'I' represents the set of items from the itemset. The association rule is formulated as shown below.

$$t(A) = \{tid \mid (tid, I) \in T, a \subseteq I \tag{1}$$

The problem studied in this paper is now as follows: for many dataset 'D', and items 'I', complex candidate itemset generation process consumes large memory and enormous execution time and therefore generates many a number of association rules.

4. ASSOCIATION RULE GENERATION

Previous works in frequent itemset mining (Zhi-Hong Deng, Sheng-Long Lv 2014) and (Zhi-Hong Deng, Sheng-Long Lv, 2015) introduced an efficient data structure set-enumeration search tree in order to decompose the mining task in smaller easily-solvable problems.

a. Association rule generation using Eclat:

The association rule generation using Eclat algorithm identifies the frequent items from large database. The advantage of using Eclat in E-ACO technique is the generation of frequent items only once whereas the conventional algorithms takes more time for identifying the frequent itemsets, necessitating to scan the database again and again, one of the most time consuming process. On the other hand, Eclat uses vertical database, through which it need scan the database only once.

Itemset '*I*'	Transaction identifier '*tid*'
i_1	{*t*20, *t*30, *t*50, *t*80)
i_2	{*t*20, *t*50, *t*60, *t*80}
i_3	{*t*30, *t*50, *t*80}
i_4	{*t*20, *t*50}

(a)

Itemset '*I*'	Transaction identifier '*tid*'
i_1, i_2	{*t*20, *t*50, *t*80)
i_1, i_3	{*t*30, *t*50, *t*80}
i_1, i_4	{*t*20, *t*50}
i_2, i_3	{*t*50, *t*80}
i_2, i_4	{*t*20, *t*50}
i_3, i_4	{*t*50}

(b)

min_sup = 2

Figure 1 (a) Vertical layout format (b) Frequent 2-itemsets in vertical layout format

With the aid of the vertical layout format, Eclat instead of explicitly listing all transactions uses an intersection property to evaluate the support of an itemset. In this way, the support of an itemset '*A*' is easily evaluated by simply intersecting '*tid*' of any two subsets '$B, C \in A$' such that '$B \cup C = A$'. Then, support of an itemset '*A*' in '*T*' symbolizes the cardinality of it '*tid*'. This implies that support of '*A*' is the number of transactions containing '*A*' in '*T*' and is as shown below.

$$_{sup}(A) = T\ (A) \tag{2}$$

Let us consider a minimum support threshold value to be 'sup_{min}'. Then an itemset '*A*' is said to be frequent if its support is not less than a predefined minimum support threshold 'sup_{min}'. Given a transaction dataset '*D*' over a training dataset '*TD*' and a minimal support threshold 'sup_{min}', set of frequent itemsets is as shown below.

$$FI\ (T, sup_{min}) = \{A \subseteq TD \mid \geq sup(A)\ sup_{min}\} \tag{3}$$

During the first scan of dataset '', for every single item a '' is maintained and items are extracted with the list of number of transactions in which they are present using intersection property.

$$t(PB) \cap d\ (PC = ((t(P) \cap t(B)) \cap (t(P) - t(C))$$

$$(t(P) \cap t) - (t(P) \cap t(B) \cap t(C))$$

$$t(PB) - t(PBC) = d(PBC) \tag{4}$$

Therefore '$n + 1$' itemset are generated from 'n' itemset by performing intersection of '*tid*' of frequent 'n' itemset.This process is repeated until no candidate itemsetare found. Once, the frequent itemsets are obtained, a correlation measure called '*Lift*' is used to generate the association rule.

$$\text{Lift}\,(\text{A, B}) = \frac{\text{Prob}(\text{A} \cup \text{B})}{\text{Prob}(\text{A}) + \text{Prob}(\text{B})} \tag{5}$$

From the results of (4), interesting association rules regarding purchasing behavior of customer is determined. Figure 3 shows the Eclat-based Frequent Itemset Generation algorithm.

Input: Transaction dataset 'D', Training dataset 'TD', Minimal support threshold 'sup_{min}', Itemset '$I = i_1, i_2, A, ..., B, ..., i_n$', Transaction '$T = t_1, t_2, ..., t_n$'
Output: efficient generation of frequent items (i.e. association rule generation)
Step 1: Begin Step 2: For all Itemset 'I' Step 3:For each transaction 'T' over 'I' Step 4:Measure support using (2) Step 5:Measure frequent itemset using (3) Step 6:Generate association rule using (5) Step 7:If $Lift$(A, B) > 1, *then* Step8: A and B are positively correlated Step 9: End if Step 10: If $Lift\ (A, B) < 1$ Step11: A and B are negatively correlated Step 12:End if Step 13:If $Lift\ (A, B) = 1$ Step14: A and B are independent Step 15: End if Step 16: End for Step 17: End for Step 18: End

Figure 2. Eclat-based Frequent Itemset Generation algorithm

The Eclat-based Frequent Itemset Generation (E-FIG) algorithm is illustrated in figure. The E-FIG takes as input a Transaction dataset '' and Training dataset ''. During each iteration, for each transaction '' over '' and for all Itemset '', support is measured and compared with the Minimal support threshold ''. Based on the result, frequent itemset is generated. An itemset is said to be frequent, if the support of the itemset is no less than the ''. The process is repeated until no frequent items can be found. With the frequent itemset generated, association rule is generated using correlation measure.

b.ACO-based optimized association rule generation

Frequent itemset mining in the market, to analyze the products frequently purchased by customer, discover customer shopping patterns, and finding the interesting rules with the derived shopping patternshelps the organization to maintain and improve sales growth using data mining techniques. This section proposes an efficient ant colony optimization algorithm to gain the customer's data to achieve customer behavioral pattern aiming at developing new business strategies. Let '$A_i(t)$' represents the number of ants at rule 'i' at time 't' and let '$TA =$ ' be the total number of ants at time 't' represents a solution '$R_k^t = \{r_{k1}^t,\ r_{k2}^t,\ \ldots,\ r_{kn}^t\}$' where '$r_{ij}^t$' is the rule that encompasses the generation of rules based on high correlated measure. As the study aims to find the optimized rule (i.e. interesting rules for the organization), the pheromone level of visited rule by ant '*k*' at time '' is updated as follows

$$\text{Pher}_k^t = (1 - \text{PER})\text{Pher}_k^{t-1} + \text{CU}(\text{R}_k^t) \tag{6}$$

Where '*PER*' is the pheromone evaporation rate, '*C*' is the correlation ratio and '$U(R_k^t)$' is the rule optimization for 'R_k^t'. The ant deposits more pheromone on a location with a higher correlation value. The pheromone level is updated only when a better solution is found where the ant instead of finding all possible rules that exist in data, the ant rather is interested in certain targets where correlation plays a key role. The ant chooses next rule '*R*' to proceed by the state transition rule as given below

$$R = \{argmax\{\alpha_{ij}(\text{t}) * f_{ij}\}, \text{ where } j \in N \tag{7}$$

Where '$\alpha_{ij}(t)$' symbolizes the intensity of pheromone trail on edge '*i, j*' at time '*t*' and '*N*' is the ant unaccomplished tour (i.e. rules) with 'f_{ij}' representing the frequency between the rules '*i, j*'.Figure 4 (a) and (b) show the vertical layout format to analyze the purchase behavior of mobiles using intensity of pheromone trail. From 4 (a) and (b), for the product '*Sony*', the high intensity pheromone trail lies at '*Processor*' that insists that whenever a customer purchases '*Sony*' there is higher probability of purchase of '*Processor*'.

Itemset '*I*'	Transaction identifier '*tid*'
i_1	{*Power Bank, Battery, Processor, Wi – Fi, Head Phone*}
i_2	{*Power Bank, Processor, Wi – Fi, Head Phone*}
i_3	{*Battery, Processor, Head Phone*}
i_4	{*Power Bank, Processor*}

(a)

Itemset '*I*'	Transaction identifier '*tid*'
i_1, i_2	{*Power Bank*, ***Processor***, *Head Phone*}
i_1, i_3	{*Battery*, ***Processor***, *Head Phone*}
i_1, i_4	{*Power Bank*, ***Processor***}
i_2, i_3	{***Processor,*** *Head Phone*}
i_2, i_4	{*Power Bank*, ***Processor***}
i_3, i_4	{***Processor***}

(b)

Figure 3(a) analyzing purchase behaviors using vertical layout format (b) analyzing purchase behavior with high correlation factor

After an ant uses the transition rule to move next based on the intensity of pheromone trail, the solution on next rule is verified so that it meets the problem constraint.

$$\sum_{i,j}^{n} = r_{ij} \le R_i \tag{8}$$

With this feasible solution, the number of rules generated that involve both frequent and rare items to the problem is obtained through which the point of sales and sales growth is achieved.

5. EXPERIMENTAL RESULTS

We have performed tests using the Eclat-based Frequent Itemset Generation algorithmusing Belgium retail market dataset. The E-ACO technique is implemented using MATLAB platform. The goal of the experimentations is(i) to verify the validity and feasibility of our E-ACO technique and (ii) to compare our algorithm results with the state-of-the-art methods, Frequent Item Sets using Equivalence Pruning (FIS-EP) [1], Frequent Item Sets using Node sets (FIS-N) [2], Eclat and Apriori.

Experimental method

Information on the dataset used is presented in table 1. The dataset originate from the Belgian retain supermarket store. The data was collected over three non-consecutive periods. The first period runs from half December 1999 to half January 2000. The second period runs from 2000 to the beginning of June 2000. The third and final period runs from the end of August 2000 to the end of November 2000. This results in approximately 5 months of data. The total amount of receipts being collected equals 88,163.

Experiment is conducted on the factors such as support, running time for frequent itemset generation, memory for association rule generation and number of rules generated.The results of the metrics of E-ACO technique is compared against the existing methods such as Frequent Item Sets using Equivalence Pruning (FIS-EP) [1], Frequent Item Sets using Node sets (FIS-N)[2], Eclat [] and Apriori [].

a. running time for frequent itemset generation

In this section, we analyze how different parameters of our algorithm affect the running time effectiveness for frequent itemset generation. The influence of '' is evaluated in figure 5 for 7 different values of ''. By '' we determine the running time taken to generate the frequent itemset and the number of frequent itemset generated with respect to the given support value.

$$RT = FI * Time(FI) \tag{9}$$

Where '*RT*' is the running time, '*FI*' symbolizes the number of frequent itemset generation and '*Time* (*FI*)' the time taken for the frequent items. The objective of this function is to allow that frequent itemsets are generated with lesser amount of time and also to extract association rules in an efficient manner. Obviously to prevent extreme cases, the running time gets reduced with an average support of 0.7. As '*min*_sup = 0.1', the running time is 145ms when E-ACO was applied when compared to the other methods that consumed higher amount of running time ranging from 175ms to 225ms. With '*min*_sup = 0.4', an optimal runtime was achieved proving the efficiency of the technique to be comparatively good.

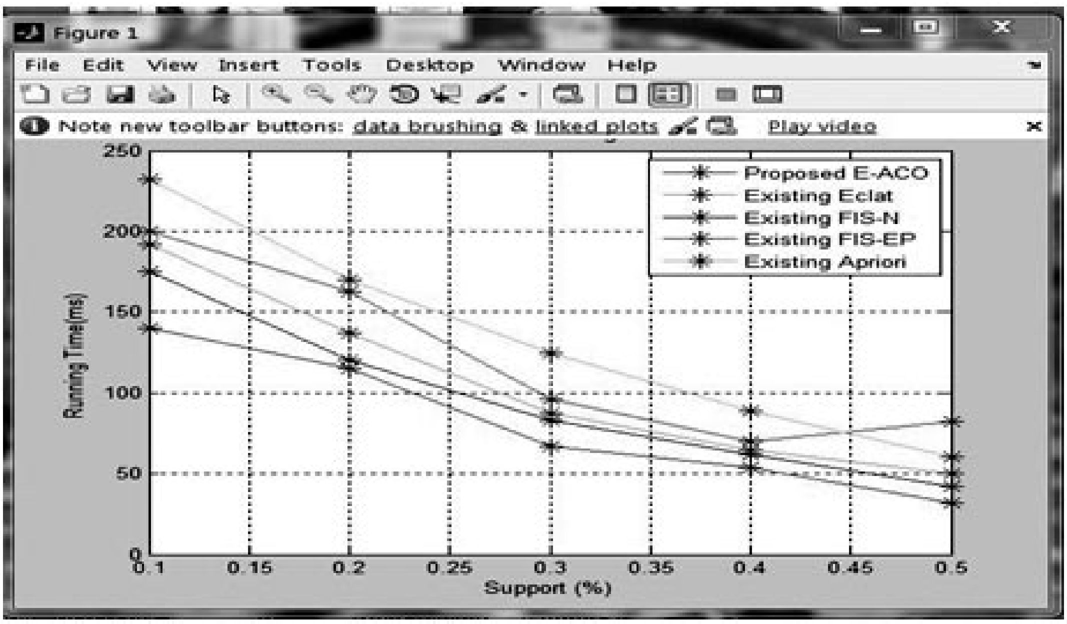

Figure 4. Performance evaluation of running time

As illustrated in the figure, the running time is reduced comparatively by applying E-ACO. This is because E-ACO uses a vertical layout format where it necessitates scanning the entire database only once, whereas the existing state of the art methods uses horizontal layout format scans the entire database each and every time resulting in the increase of the execution time. On the other hand by applying Eclat algorithm in E-ACO technique, support count information is obtained from '*k*' itemsets without scanning the entire database. This advantage over the other methods results in minimizing the runtime for frequent itemset generated using E-ACO technique by 16.57% compared to FIS-EP, 28.55% compared to FIS-N, 42.26% compared to ECLAT and 42.26% compared to Apriori.

b. Memory for association rule generation:

In the following we briefly analyze the memory for association rule generation. In figure 6, we use the Belgium retail market dataset and determine for each support value, the memory consumption for each frequent itemset generation method. The experiment indicates that the measure of memory for association rule generation using vertical layout format with the aid of intersection property reduces the memory consumption when compared to the existing state-of-the-art methods. By increasing the support measure between time intervals, memory is reduced from one support measure to the other. The aim of measuring the memory consumption is not just to measure the consumption but to reduce it to identify different types of evolution and development while analyzing the purchase behavior of customer.

$$M = FI * M\,(FI) \tag{10}$$

Where '*M*' is the memory consumed for association rule generation, '*FI*' symbolizes the number of frequent itemset generated and '*M*(*FI*)' the memory required for association rule generation.

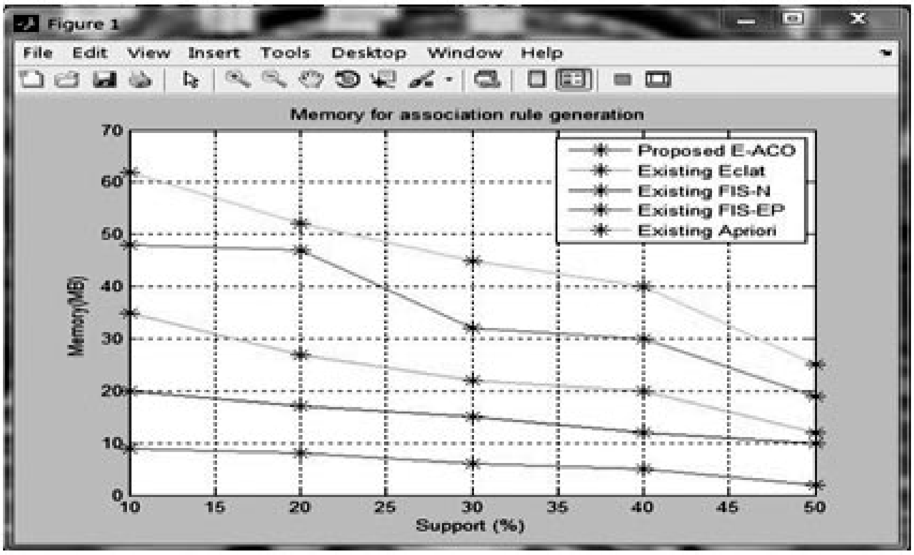

Figure 5. Performance measure of memory with respect to different confidence measure

The performance measures of memory consumption using E-ACO, FIS-EP, FIS-N, Eclat, Apriori is evaluated in figure 6. We analyze the influence on the memory consumption for generating association rules with a varying support value on the '*x axis*'. The higher the support value, lesser is the memory consumption using all the methods. But comparative analysis performed and implemented in JAVA shows performance gain of E-ACO by 23.04% compared to FIS-EP, 42.40% compared to FIS-N, 61.75% compared to ECLAT and 80.64% compared to Apriori. Therefore, we can draw the following conclusions regarding memory for association rule generation. By applying a correlation measure called '*Lift*', correct amount of association rules are generated and thus results in reduced memory consumption.

a. Rules generation:

In the following, we analyze whether our technique is able to optimize the rules for different settings. Up to now, we used our association rule generation using Eclatfor generating frequent itemsets and extracting association rules that utilizes the vertical layout format and the correlation measure based on the lift function. Now, we additionally apply the ACO technique with a variant that performs optimization with the rule generated using Eclat. In Figure 7, the

Belgium retail market dataset is used and we determine for each intensities of pheromone trail the number of optimized association rules generated. The experiment indicates that the number of number of rules generated for the approach without vertical layout format is larger than for our enhanced E-ACO technique.

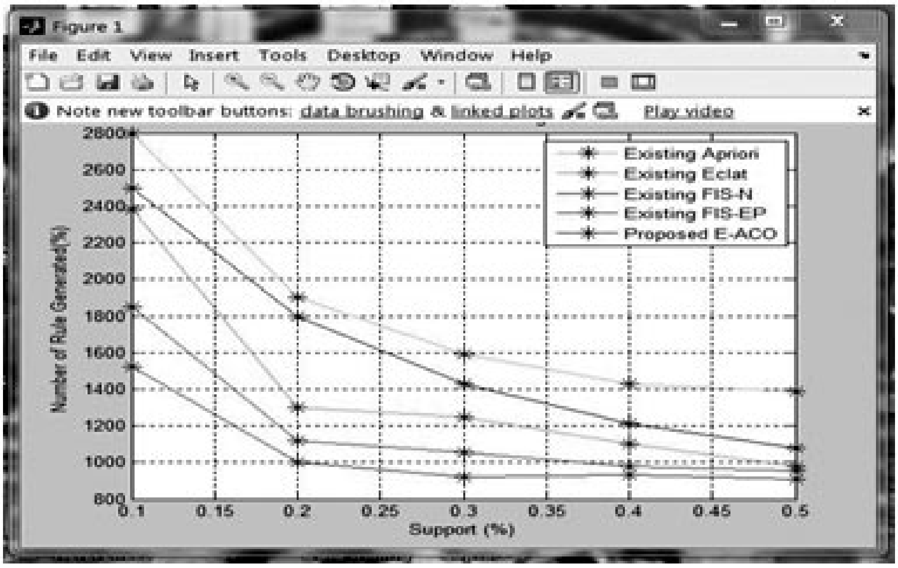

Figure 7. Performance measure of number of rules generated

As shown in the figure, the aim of E-ACO technique is not just to obtain minimum number of rules but also to extract optimal number of rules for determining the customer behavior. In Figure 7, we plot the number of optimal number of rules generated and the four state-of-the-art works compared with the E-ACO technique over different support measure. The number of rules generated is minimized by using E-ACO technique because of the correlation ratio obtained through the Eclat and then deriving the intensity of pheromone trail. This in turn optimizes the rules generated using E-ACO technique by 11.70% compared to FIS-EP, 21.48% compared to FIS-N, 36.18% compared to Eclat and 52.92% compared to Apriori.

6. CONCLUSION:

In this article, we projected a technique for minimizing the frequent itemset and association rule generation and sodevelop the point of sales in supermarkets. In compare to existing methods, we use vertical layout format optimize the rules generated based on the correlation measure, that is, rules are not only generated based on the correlation measure but also on the intensity of correlation measure. Therefore, our technique is especially suited for analyzing the customer behavior regarding purchase of products and improves the point of sales in supermarkets. We have taken datasets in Belgian retail supermarket. . For maximal elasticity, our tracing frequent itemset mining technique distinguishes several state-of-the-art methods. In the investigational evaluation we showed that time and memory required for frequent itemset generation is reduced through optimized generation of rules.

REFERENCES:

[1] Zhi-Hong Deng , Sheng-Long Lv, “PrePost+: An efficient N-lists-based algorithm for mining frequent itemsets via Children–Parent Equivalence pruning”, *Elsevier, Expert Systems with Applications*, **42**, **13**, 2015.

[2] Zhi-Hong Deng, Sheng-Long Lv, “Fast mining frequent itemsets using Node sets”, *Elsevier, Expert Systems with Applications,* **41**,**10**, , 4505–4512, 2014.

[3] Michalis Mavrovouniotis, Shengxiang Yang, “Ant colony optimization with immigrants schemes for the dynamic travelling salesman problem with traffic factors”, *Elsevier, Applied Soft Computing,*13,10, , 4023–4037, 2013.

[4] Chao-Yang Pang, Ben-Qiong Hu, Jie Zhang, Wei Hu, and Zheng-Chao Shan, “Applying Data Clustering Feature to Speed Up Ant Colony Optimization”, *Hindawi Publishing Corporation, Abstract and Applied Analysis*,**2014**, 1-9, 2014.

[5] Xin Chen, Yongquan Zhou, and Qifang Luo, “A Hybrid Monkey Search Algorithm for Clustering Analysis”, *Hindawi Publishing Corporation, The Scientific World Journal*, **2014**,1-17, 2014.

[6] AndrásKirály, Attila Gyenesei, and JánosAbonyi, “Bit-Table Based Biclustering and Frequent Closed Itemset Mining in High-Dimensional Binary Data”, *Hindawi Publishing Corporation, The Scientific World Journal,***2014**, 1-8,2014.

[7] Yi Zeng, Shiqun Yin, Jiangyue Liu, andMiao Zhang, “Research of Improved FP-Growth Algorithm in Association Rules Mining”, *Hindawi Publishing Corporation, Scientific Programming,***2015**,1-7, 2015.

[8] Adam Perer ,Fei Wang, Jianying Hu, “Mining and exploring care pathways from electronic medical records with visual analytics”, *Elsevier, Journal of Biomedical Informatics* **56**, , 369–378, 2015.

[9] TamirTassa, “Secure Mining of Association Rules in Horizontally Distributed Databases”, *IEEE Transactions on Knowledge and Data Engineering,* **26**, **4**, 970-983, 2014.

[10] Yan Zhang, Fan Zhang, ZhemingJin, Jason D. Bakos, “An FPGA-Based Accelerator for Frequent Itemset Mining”, *ACM Transactions on Reconfigurable Technology and Systems (TRETS),* **6**,**1**,1-17, 2013.

[11] Eric Hsueh-Chan Lu, Vincent S. Tseng, and Philip S. Yu, “Mining Cluster-Based Temporal Mobile Sequential Patterns in Location-Based Service Environments”, *IEEE Transactions on Knowledge and Data Engineering,***23**, **6**, 914-927, 2011.

[12] Osvaldo Navarro, Rene´ Cumplido, Luis Villasen or-Pineda, Claudia Feregrino-Uribe, Jesu´ s Ariel Carrasco-Ochoa, “A Node Linkage Approach for Sequential Pattern Mining”, *PLOS ONE*, **9, 6,**1-16, 2014.

[13] SarraSenhadji, Salim Khiat, and HafidaBelbachir, “Association Rule Mining and Load Balancing Strategy in Grid Systems”, *The International Arab Journal of Information Technology,***11**, **4**, 338-344, 2014.

[14] VimalDhanasekar and TamilarasiAngamuthu, “An Efficient Approach for Effectual Mining of Relational Patterns from Multi-Relational Database”.

[15] PranayAnchuri , Mohammed J. Zaki, Omer Barkol ,Ruth Bergman ,Yifat Felder Shahar Golan , Arik Sityon, “Graph mining for discovering infrastructure patterns in configuration management databases”, *Springer, Knowledge and Information Systems,***33, 3,**491-522, 2012.

[16] Daniel M. Bartels, Eric J. Johnson, “Connecting cognition and consumer choice”, *Elsevier, Cognition,***135,**47–51, 2015.

[17] Fan Zhang, Yan Zhang, Jason D. Bakos, “Accelerating frequent itemset mining on graphics processing units”, *Springer, The Journal of Supercomputing,***66,1**, 94-117, 2013.

[18] Bay Vo ,Tzung-Pei Hong , Bac Le, “DBV-Miner: A Dynamic Bit-Vector approach for fast mining frequent closed itemsets”, *Elsevier, Expert Systems with Applications,***39, 8,**7196–7206, 2012.

[19] ToonCalders, NeleDexters ,JorisJ.M.Gillis , BartGoethals, “Mining frequent itemsets in a stream”, *Elsevier, Information Systems,***39**, 233–255,2014.

[20] Zhong-jie Zhang, Jian Huang, and YingWei, “FI-FG: Frequent Item Sets Mining from Datasets with High Number of Transactions by Granular Computing and Fuzzy Set Theory”, *Hindawi Publishing Corporation, Mathematical Problems in Engineering,***2015,** 1-15,2015.

[21] M.Sathya, Dr.K.Thangadurai, “Market Basket Analysis in Association Rule Mining Based on Frequent Itemset” *International Journal of Applied Engineering Research,*0973-4562 10**, 82** (2015).