

## Analysis of Rule Generation Techniques using Computational Method

Maragatham, G. and M. Lakshmi  
Department of Computer Science and Engineering,  
Sathyabama University, Chennai-119, India

---

**Abstract: Problem statement:** Enhanced decision making seeks a right of way mainly in business sectors for mining profitable patterns in databases. To address this problem, the authors have incorporated the temporal and utility concept of mining effective rules. Though the objective of generating profitable rule set is achieved. Still in effective decision making, Optimization of the rule set is required. Therefore the authors have used Particle Swarm Optimization technique for optimizing the generated rule set. To optimize the rule set, the authors have introduced weighted fitness function, which is based on the concept of weighted support and weighted confidence. **Approach:** In this article the authors have done a meticulous performance study of the proposed approach Optimized UTARM also called UTARM\_PSO. Also, comparative analysis of the proposed approach with the apriori\_PSO technique is done. **Results and Conclusion:** The experimental results show that the our approach has performed the optimization consistently and precisely on values of support ranging from 80-90%. The Comparative analysis pointed to a 15% in the overall performance of our approach against the existing approach.

**Key words:** Utility based temporal association rules, particle swarm optimization, weighted fitness function

---

### INTRODUCTION

The ultimate intention of the study is to analyze the behavior of our proposed study with the existing rule generation approach. Comprehensive reviews over the literature accelerate us to develop a technique for association rules using temporal and utility. By taking into account the cost of database scans, we have adapted the technique addressed by (Lee *et al.*, 2003) for mining association rules with reduced database scans. The technique suggests a model for mining association rules based on temporal factors from partitioned database. For mining profitable item sets, we need utility concept. Utilities are semantics which provide information about an expediency property of an item. Adding this feature to the items in the partitioned databases adds better value for mining profitable items. Mining of association rules by means of utility factor is one of the issues in pattern mining. Based on this constraint, the concept of utility based temporal association rules also called UTARM technique is proposed (Maragatham and Lakshmi, 2010). When the rules tend to be more, decision making may be lengthier. To overcome this snag, Optimization technique is required. Particle swarm optimization Technique is availed for optimization by researchers (Ykhlef, 2011). We have

used Particle swarm optimization technique for reducing the rule set with modified fitness function.

The modified fitness function is based on the weighted support and weighted confidence value and finally the proposed technique is named as UTARM\_PSO also called as weighted particle swarm optimization technique (Maragatham and Lakshmi, 2011). In this editorial the meticulous study on the performance analysis of the approach is done on varied aspects such as the number of rules generated, execution time consumed and memory usage. For validating our approach, a comparative analysis is done with respect to the existing approach.

### MATERIALS AND METHODS

The proposed approach is subjected for the experimental evaluation for analyzing the performance. The implementation uses Java-JDK 1.6.0 and the experimental setup includes the PC running on core i3 processor with 3 GB RAM. The process of assessment of optimized approach uses two data sets. The details of the data set are.

**Retail data set:** It is a real time data set, which is described by "Tom Brijs". We have considered a fragment

---

**Corresponding Author:** Maragatham, G., Department of Computer Science and Engineering, Sathyabama University, Chennai-119, India

of the retail data set for the evaluation of our proposed approach. The portion selected is divided into a number of partitions and tested for the experimental evaluation.

**T10I4D100K data set:** The data set is a synthetic type from FDIMDR. A Fraction of the data set is subjected to the evaluation process.

**UTARM technique:** Initially the rules are generated by this technique. For tumbling the total number of database scans, scan diminution or reduction technique (Lee *et al.*, 2003) is used. This technique is triggered with utility mining for mining utility based temporal association rules. In the beginning, we offer input database D as a set of partitions,  $D = \{P_1, P_2, P_3, \dots, P_n\}$ , Where each partition has a set of transactions  $T = \{T_1, T_2, \dots, T_s\}$  and has a set of items,  $X_D = \{x_1, x_2, \dots, x_m\}$ . Each item in the partition  $P_k$  contains different utility values. The following shows the steps in generating rules:

- Initially, Partition one is considered and all the combinations of frequencies 2-item sets are found

- Secondly, With the items generated in the previous step, Frequent Temporal Utility item sets are identified using the utility table and minimum support and minimum confidence criteria
- Next using Partition two, the candidate 2-item sets are recognized and Frequent Temporal Utility 2 – item sets from Partitions one and two ( $P_1 + P_2$ ) are identified
- The previous step is repeated for all the partitions from Partition one to Partition ‘n’ for mining Frequent Temporal Utility 2-item sets
- Next by using Frequent Temporal Utility 2 – item sets all Frequent Temporal Utility 1 item sets are found
- Using the temporal concept, the two item sets are merged to form 3 item sets. In this way higher order item sets are generated
- Finally Utility based temporal association rules are generated based on minimum confidence adopted

These rules are considered as input to the UTARM\_PSO method for optimization.

```

Algorithm: UTARM_PSO (T, Rs, r, Ffitness)
Input : Set of transactions
Output : Optimized set of rules.

Start
Select Transactions set T
For each transactions t ∈ T
    Call UTARM (t, U, Support, Confidence, R)
    Return(R)
End of for loop
For each r ∈ R
    Start PSO
    Initialize velocity, r, Rs // r = particle, Rs = Swarm
    Now
    Update velocity, Position X
    For each r
         $velocity(r_i)^{new} = velocity(r_i)^{old} + r_{p_{best}}(best\_position(r_i) - x_i) + r_{g_{best}}(swarm\_position - x_i)$ 
         $x_i^{new} = x_i^{old} + velocity(r_i^{new})$ 
    End of for loop
    Call Ffitness
     $F_{fitness} = X_i \times Conf(r) \times \log(FTU\_Support(r))$ 

    If (Ffitness > Fthreshold) // Fthreshold = Threshold set for the fitness value
        Select r
    Else
    Discard r
End of for loop
End
    
```

Fig. 1: Pseudocode of proposed approach-Optimized UTARM Technique  
1023

**Optimized UTARM approach:** The Rules generated from the above technique is used as an input for this method and PSO technique with modified fitness function is used for rule optimization. In Fig. 1. The fitness function comprises of confidence and support factors. The confidence and support factors are based on the temporal and utility factors. Hence, the fitness function is called as weighted fitness function. The final output is the optimized set of rules. Figure 1 shows the Pseudocode of the weighted PSO approach.

obtained from the experimental evaluation of the retail data set with Optimized approach. The data set is provided varyingly on the size of the population and various support values, the results are mapped in accordance with the behavior of optimized UTARM approach to the differential data. Considering the mapping, it is found that the response of this approach is steadfast in higher support values. While analyzing the Fig. 3, the execution time is consistently varied for different population size.

**RESULTS**

The performance of the optimized UTARM approach is evaluated in the midst of synthetic and real data set. The data set is analyzed in different stages of appraisal for obtaining a perfect study of the technique. The analysis includes the evaluation of retail data set under diverse conditions. The authors have provided the data set as different partitions, each partition containing 500 transactions each. The transactions are evaluated by varying the support values, the values are ranging from 60-90. The performance evaluation at different stages are plotted.

**Analysis based on No. of iterations with Retail data set:** The main features that legalize the optimization concept are the number of iterations, which is obligatory for the processing. The mappings of the performance of the approach over different iterations are analyzed. Here also, the authors use a changeable population size approach for the different supports in order to achieve a serious evaluation. In the case of a number of Iteration, the optimal results are obtained when the number of iterations is triumph to comparably high values. The proposed optimized approach showed the most optimal solution, for a support of 80% and the iteration of 100 are set. The Fig. 4 and 5 represents the performance of our optimized approach evaluated under a different number of iterations with respect to rules generated and execution time.

**Analysis based on population size with Retail data set:** The Fig. 2 and 3 are plotted based on the readings

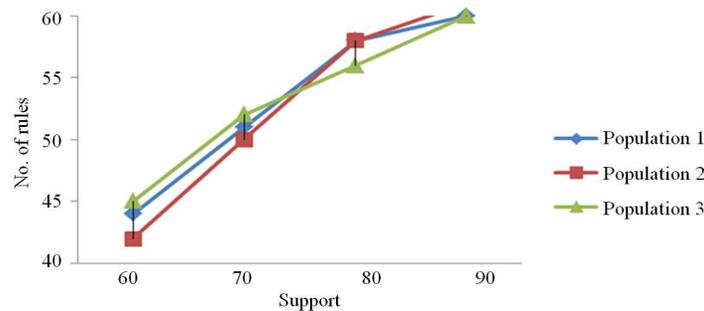


Fig. 2: Evaluation based on the number of rules for different population size-proposed approach on Retail Dataset

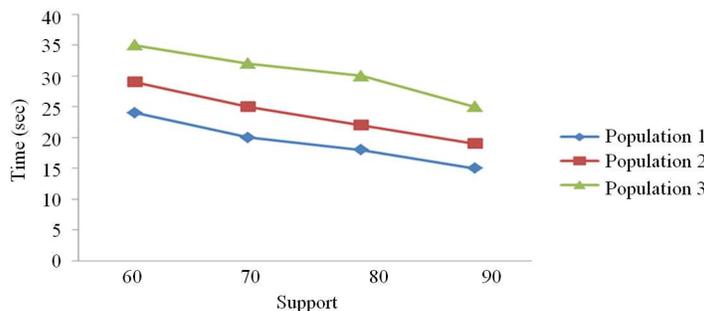


Fig. 3: Evaluation based on execution time for different Population-proposed approach on Retail Dataset

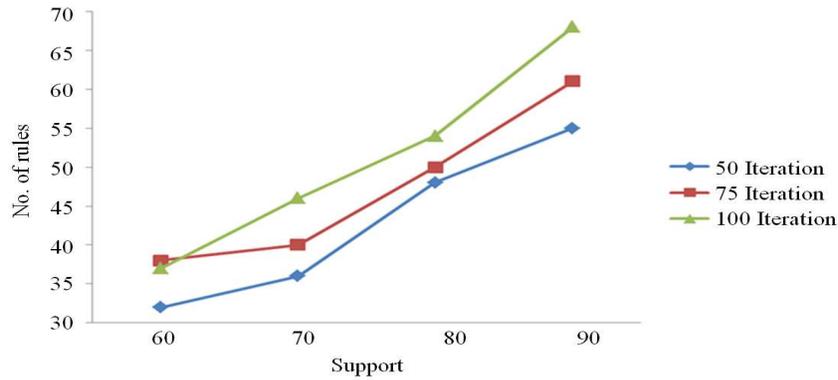


Fig. 4: Evaluation based on the number of rules for different iterations-proposed approach on Retail Dataset

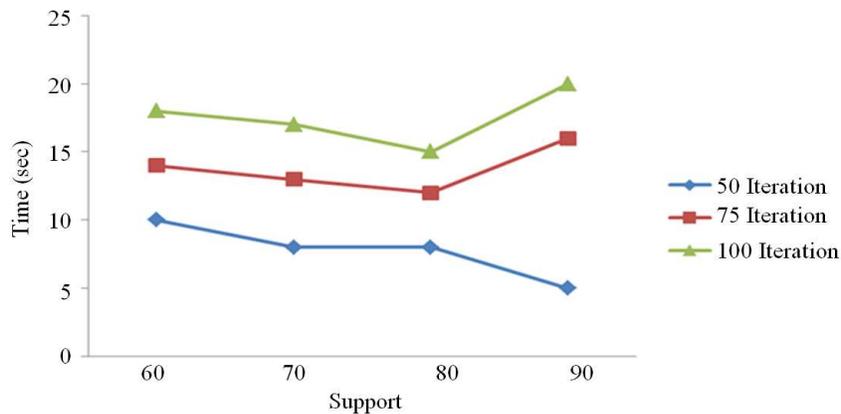


Fig. 5: Evaluation based on execution time for different iterations-proposed approach on Retail Dataset

**Comparative analysis of UTARM\_PSO and Apriori\_PSO:** The data sets for instance T10I4D100K and Retail are used for the comparative analysis. For the comparative analysis the authors considered 1000 transactions for each of the data sets. Each data set is tested by varying support measures. The value ranges from 60-90 under a constant confidence value.

**Analysis based on T10I4D100K:** T10I4D100K, which is a synthetic data set, is considered in the comparative analysis. The comparative analysis deals with criteria such as the number of rules generated, execution time and the memory usage. The details of the comparison are given in Fig. 6-8. The Fig. 6 shows plotted graph of the number of rules obtained while optimizing the proposed approach and the existing approach. Figure 7, gives the comparison of execution time that is required for both techniques for generating the expected result. The Fig. 8 represents the memory usage details supported by the techniques. From the mappings it is evident that, the performance of the proposed approach

is better than the existing approach at higher support values. The traditional approach gives a consistent performance to different support values; on the other hand, the proposed approach is showing consistency and accuracy at higher support values, which in turn increases the efficiency of this approach. The time and space requirement is also comparably in a lower degree for our approach against the existing approach. The analysis is given in Table 1. (UTARM + Optim stands for UTARM technique using optimization. Prior + Optim implies a priori technique using optimization )

**Analysis based on Retail data set:** Retail, which is a real-time data set, is considered in the comparative analysis. The comparative analysis deals with diverse criteria such as the number of rules generated, execution time and the memory usage. The details of the comparison are plotted in Fig. 9-11. Findings are summarized in Table 2. (UTARM + Optim stands for UTARM technique using optimization. Prior + Optim implies a priori technique using optimization)

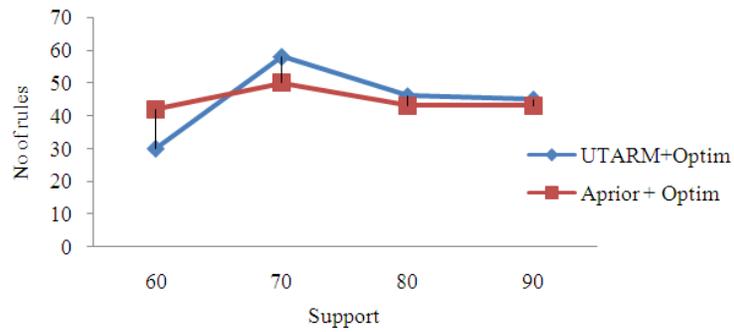


Fig. 6: Analysis based on the number of rules with T10I4D100K

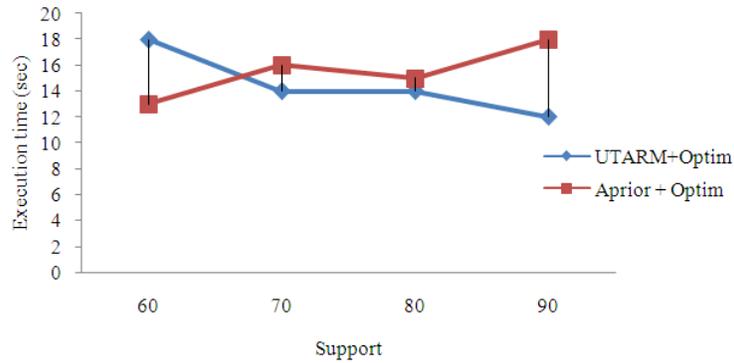


Fig. 7: Analysis based on execution time with T10I4D100K

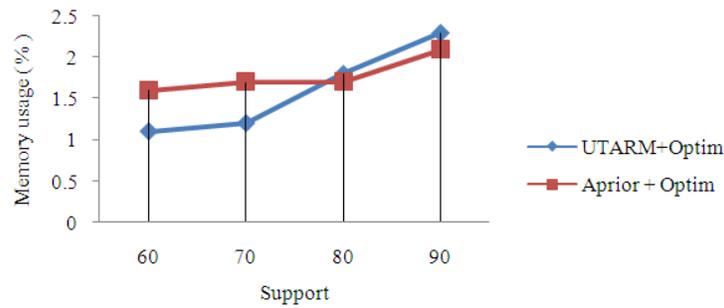


Fig. 8: Analysis based on memory usage with T10I4D100K

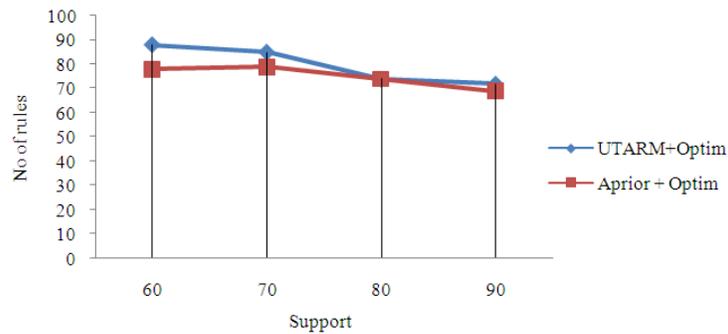


Fig. 9: Analysis based on the number of rules with Retail  
1026

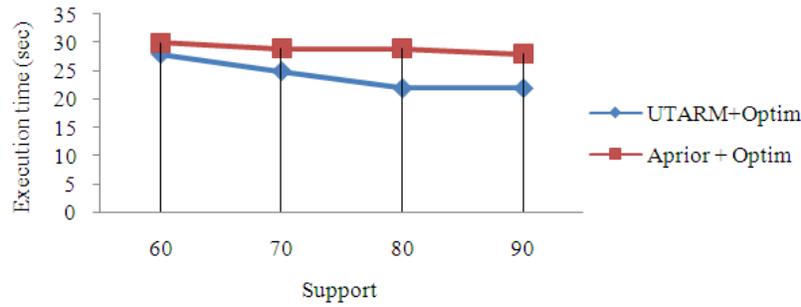


Fig. 10: Analysis based on execution time with Retail

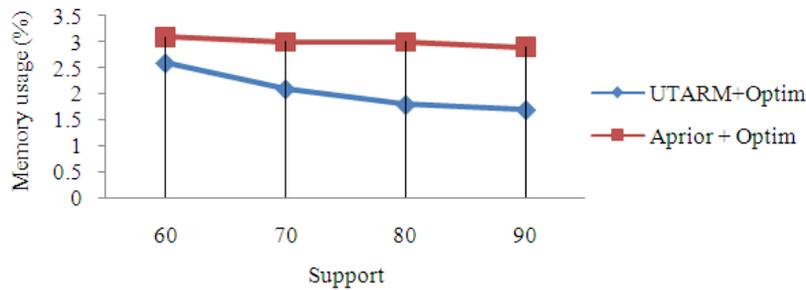


Fig. 11: Analysis based on memory usage with Retail

Table 1: Analysis of T10I4D100K

Support	No. of Rules		Execution Time (sec)		Memory usage (%)	
	UTARM +optim	Aprior +optim	UTARM +optim	Aprior +optim	UTARM +optim	Aprior +optim
60	30	42	18	13	1.1	1.6
70	58	50	14	16	1.2	1.7
80	46	43	14	15	1.8	1.7
90	45	43	12	18	2.3	2.1

Table 2: Analysis of retail data set

Support	No. of Rules		Execution Time (sec)		Memory usage (%)	
	UTARM +optim	Aprior +optim	UTARM +optim	Aprior +optim	UTARM +optim	Aprior +optim
60	88	78	28	30	2.6	3.1
70	85	79	25	29	2.1	3.0
80	74	74	22	29	1.8	3.0
90	72	69	22	28	1.7	2.9

**DISCUSSION**

The proposed technique and existing approaches are analyzed in facet with respect to both data sets (real and synthetic). The analysis shows proposed approach has ascended over the existing technique in the case of both data sets. The Fig. 6-8 shows the behavior of proposed and existing approaches on synthetic data set. Fig. 9-11 shows the behavior of proposed and existing approaches with respect to the retail data set. Finally,

the analysis concludes that our approach outperforms the existing approach using optimization in various aspects such as the number of rules to be optimized, execution time and memory usage. The details are specified in Table 1 and 2.

**CONCLUSION**

The performance of the proposed technique is encouraging in differentiating the redundant rules and optimization. The result of this approach is done with the help of the real-time data sets and synthetic data sets. The evaluation of the result shows that the proposed technique optimizes the rules which are generated from the UTARM algorithm, efficiently. The extensive comparison study reveals our optimized approach is 15% added accurate and precise than the existing technique.

**REFERENCES**

Lee, C.H., M.S. Chen and C.R. Lin, 2003. Progressive partition miner: An efficient algorithm for mining general temporal association rules. IEEE Trans. Knowl. Data Eng., 15: 1004-1017. DOI: 10.1109/TKDE.2003.1209015

- Maragatham, G. and M. Lakshmi, 2010. A strategy for mining utility based temporal association rules. *Trendz Inform. Sci. Comput.* DOI: 10.1109/TISC.2010.5714604
- Maragatham, G. and M. Lakshmi, 2011. A weighted particle swarm optimization technique for optimizing association rules. *Proceedings of the 4th International Conference on Recent trends in Computing, Communication and Information Technologies*, Dec. 9-11, VIT University, Vellore, India, pp: 675-684.
- Ykhlef, M., 2011. A Quantum Swarm Evolutionary algorithm for mining association rules in large databases. *J. King Saud Univ. Comput. Inform. Sci.*, 23: 1-6. DOI: 10.1016/j.jksuci. 2010.03.001