IASET: Journal of Computer Science and Engineering (IASET: JCSE)
ISSN(P): Applied; ISSN(E): Applied
Vol. 1, Issue 1, Jan - Jun 2016, 31-38

International Academy of Science, Engineering and Technology
Connecting Researchers; Nurturing Innovations

AN EFFECTIVE AND EFFICIENT METHODOLOGY TO REDUCE SPACE CONSTRAINT IN OPTIMIZED ASSOCIATION RULE MINING

BHARATHI T[1] & KRISHNAKUMARI P[2]

[1]Research Scholar, RVS College of Arts and Science, Coimbatore, Tamil Nadu, India

[2]Director of MCA, RVS College of Arts and Science, Coimbatore, Tamil Nadu, India

ABSTRACT

Association rule learning is a well researched and well explored approach to determining the interesting relatives in high dimensional data along with variables. It analyzes and gives the strong rules discovered in databases by means of diverse measures of interestingness. On the other hand, find out the threshold values of support and confidence is critically affect the association rule mining quality or accuracy. In the previous works, proposed an optimization technique such as PSO, Modified PSO and Modified AFSA to overcome these problems. Even though in those system memory consraints are left. This is the considerable disadvantage in the system. Most of the algorithms consume more space, generate many candidate item sets, unnecessary items exists, primary clustering and collision occurs in the hash based linear probing method, it takes more time to find a frequent item sets, etc. In order to reduce the space constraint as well as to address scalability, we use Cuckoo Hashing which is also one of the open addressing techniques. Additionally with the intension of increase the accuracy of the system and further reduce the space constraint we are introduce the rule pruning strategy using branch and bound algorithm. From the experimentation result on a dataset shown that the proposed system is undoubtedly reduce the memory constraints as well as it will be used for increasing the accuracy of the system.

KEYWORDS: Association Rule Mining, Branch and Bound, Frequent Item Sets, Hashing and Rule Pruning Strategy

INTRODUCTION

Data mining is a promising approach to address the difficulty of reconstructing data into useful knowledge information from the user who can mine the results which they want. These rules are builded based on the knowledge by data mining techniques in which one of most challenging steps in an association rule is detection procedure knowledge validation. To add tress this issue association rules have been extensively used in several application areas for determining patterns in data and to create the association rules. The pattern discloses combinations of events that happen at the same time according to the interesting associations in the middle of a large set of data items. The attributes value conditions will be shown by association rule that arise often together in a given dataset. Without including the antecedent ("if" element) and the consequent ("then" element), an association rule has two important parameters such that Support and Confidence which are express the rate of ambiguity about the rule.

The most delegate association rule technique is the Apriori algorithm which is continually creates candidate item sets and uses minimal support and minimal confidence to prune these candidate item sets to obtain high-frequency item sets. Because the processing time of the Apriori algorithm is high, very important issue is a computational complexity. To address this problem in Apriori, many researchers have proposed ebhanceed association rule-related algorithms. Toivonen proposed the sampling algorithm in 1996 [1]. This algorithm is involved in determining association rules to

decrease database activity. The technique produces exact association rules, but in some cases some missing association rules might exist i.e., it does not generate all the association rules. The DIC algorithm was proposed by Brin et al. [2], DIC partitions a database into a number of blocks and then it is frequently scans the database. Genetic algorithms have also been applied in ARM [3]. These genetic approaches can produce appropriate threshold values for ARM. In another research, an ant colony system was also proposed under multi-dimensional constraints [4] which are incorporated with the clustering technique to give more accurate rules [5]. After that particle swarm optimization is proposed in R.J. Kuo [6], but in this algorithm feasible problem is occurred.

Most basic particle swarm optimization algorithm analysed for the optimum fitness value of each particle and next determines corresponding support and confidence as minimal threshold values consequent to the data are distorted into binary values. To enhance the feasibility of the system the modified PSO (Particle Swarm Optimization) [7] is proposed to offer the feasible values. The modified PSO has a number of swarm population sizes, the number of highest generation, and three predetermined parameters will be determined Cw, Cp and Cg. In each iteration, the particle's position value in all extents will be kept or be restructured by its pbest value or be restructured by the gbest value or returnd by generating a new random number.

Finally in the preliminary work, to improve the feasibility of the process the modified fish swarm optimization is developed and it is used to provide the feasible values. The MAFSA (Modified Artificial Fish Swarm Algorithm) is an approach based on swarm behaviors that was motivated from social behaviors of fish swarm in the nature. AFs search the problem space by basic behaviors. AF lives in the background which significantly is solution space and other AF's domain. Food density rate in water area is considered as an AFSA objective function. At last, AFs attain to a point which its food density rate is high (global optimum). In this algorithm basic behavior of AFSA is changed. The basic behaviors of AFSA are prey, follow and swarm.

RELATED WORKS

The idea of association rule cover for pruning association rules is introduced in [8]. A cover is fundamentally a subset of the exposed associations that can cover the database. The number of rules in a cover can be fairly small. A greedy algorithm is proposed to obtain a good cover after that the remaining rules are pruned. The issue with this approach is that the benefit of association rules, its completeness, is lost. Undoubtedly, a good method is to summarize the discovered rules. From this summary, the user can get an overall picture of the domain. In [9], proposed a rule pruning method using *minimum improvement*, which is the difference between the confidence of a rule and the confidence of any proper subrule with the same resultant. If the particular rule doesn't meet this minimum improvement in confidence then that will be pruned. In many researches, several techniques are proposed for rule pruning such as pessimistic error rate [10] and minimum description length based pruning [11].

Hashing, first illustrated in survey literature by Dumey [12], appeard in the 1950s as a space efficient technique for speedy retrieval of information in sparse tables. Latest approach called Two-Way Chaining [13] will also be analysed. Carter and Wegman [14] achieved in the case of removing randomness assumptions from the study of Chained Hashing, proposed the idea of universal hash function families. By using Carter and Wegman's universal family a random function is implemented, at that time chained hashing has constant time for every dictionary operation. Linear Probing and Double Hashing provably provide with convincing information the above performance bounds by using the hash function family of Siegel [15], [16] respectively.

EXISTING METHODOLOGY

Modified Artificial Fish Swarm Optimization was proposed in preliminary work for association rule mining. This work contains two parts that are preprocessing and mining. The first part procedure is related to calculating the fitness values of AFSA. Thus, the data are transformed and stored in a binary format. Then, the swarm's search range is set using the Item set Range (IR) value.

The fitness value of each particle calculated from the fitness function.

Fitness (k) = confidence (k) log (support (k) length (k) + 1)

Where, Fitness (k) is the fitness value of association rule type k. Confidence (k) is the confidence of association rule type k. Support (k) is the actual support of association rule type k. Length (k) is the length of association rule type k.

The second part of the algorithm provides the main contribution of this study; the modified AFSA algorithm is employed to mine the association rules. First, we proceed with AFSA encoding, this step is similar to chromosome encoding of genetic algorithms. Generate a population of AFSA according to the calculated fitness value is the next step. Finally, the AFSA searching procedure proceeds until the condition is reached i.e., the best AF is found. The minimal support and minimal confidence are represented by the support and confidence of the best AF. So, we can use this minimal support and minimal confidence for further association rule mining.

PROPOSED METHODOLOGY

To reduce the memory constraints in the system and to improve the accuracy of the system we are proposing the hashing function and rule pruning strategies. By using the cuckoo hashing function we can reduce the memory constraint that means memory storage in our system. Rule pruning is used for further reduce the space constraints as well as it will upgrade the accuracy rate of the system. In the following section we are shown the detailed content about our proposed methodology.

Hashing Technique

The limitations of traditional Apriori algorithm are step by step reduced by hashing approach [17]. Hashing method is the very well-organized approach in searching to the exact data item set in a little time. Hashing is the process which places each data item at the index of memory location. The two fundamental things needed for hashing approach. They are hash table and hash functions.

A hash table is builded of two parts: an array (the searched data is stored in the real table) and a mapping function, which is referred as a hash function. The hash function offers a method for assigning numbers to the input data in that manner of the data can then be accumulated in the array index related to the assigned number. Hash table method is primarily used to decrease the size of candidate item sets. The item sets are mapped into the different buckets of a hash table according to the hash functions, and enlarge the respective bucket counts. If the bucket count of particular ietm set is less than the minimum support then that is removed from the candidate set.

We are used the Cuckoo hashing method which can be used to improve performance of hashing among many techniques. The hash table is split into two smaller tables in Cuckoo hashing technique. And those two smaller tables have equal size, and also each hash function offers an index into one of these two tables. Generally, this hashing is defined as a

dynamization of a static dictionary. This new dictionary consists of two hash tables such that T_1 and T_2 each contain r words, and two hash functions h_1, h_2. Every key $x \in S$ is stored either in cell $h_1(x)$ of T_1 or in cell $h_2(x)$ of T_2, but not at all in both. Lookup function of our hashing is defined as,

function lookup*(x)*

return $T_1[h_1(x)] = x \vee T_2[h_2(x)] = x$

end

The thought is to load elements around until an element is placed in a vacant position (with content).

An insertion of x starts by putting x in T1, kicking out any element that might reside in T1 [h1(x)], making it nestless. If there is a nestless key, it is inserted in T2 in the same fashion, and presently using the circular order 1, 2, 3, 1, 2, 3,.....In case the number of element moves go beyonds a threshold Max Loop the process gives the element which is presently nestless. The pseudo code of procedure is as follows (where is used to denote the swapping of two variable values).

procedure insert (x)

if lookup (x) then return

loop MaxLoop times

x ↔ T1 [h1(x)]

if x = ⊥ then return

x ↔ T2 [h2(x)]

if x = ⊥ then return

end loop

rehash (); insert (x)

end

Finally, consider the cost of rehashing process. At first we consider only *forced* rehashes, origined by failed insertions. These happen while the insertion loop runs for t = MaxLoop iterations.

Rule Pruning Approach

The huge number of Association rules or patterns is builded from high dimensional data. However the majority of the association rules have redundant information and therefore that rules can't able to use for an application directly. So pruning rules is used to obtain very important rules or knowledge. In our proposed system, we are proposing a novel rule pruning approach which is used the branch and bound technique to perform pruning task.

Branch-and-Bound technique is a common search method. It is an approach which is regularly implemented for obtaining the optimal solutions. Generally it is applied while the greedy and dynamic programming approaches fail. This technique is begins from the best estimated solution, in the root of the B&B tree, and tries to find the best exact one.

In our system, we used the B&B for the rule pruning purpose. For this we are considering the objective function or optimized solution as threshold for prune the rule in ARM. We are taking confidence as the objective function.

Algorithm 1: Rule Pruning Using Branch and Bound Approach

begin

activeset: ={rules};

bestval: = minConf;

while activeset is not empty **do**

choose a branching node, node k € activeset;

remove node k from active set;

generate the children of node k, child i, i=1,...,n_k,

and corresponding optimistic bounds $conf_i$;

for i=1 to n_k **do**

if $conf_i$ worse than bestval **then** kill or prune child i;

else if child is a complete solution **then**

add child i to bestRuleset

end if

end for

end while

end

In the above algorithm, rule pruning process performed according to the B&B technique. Firstly, we are taken the all rules in the active set and assign the minConf as a best value or threshold i.e., objective function for the rule pruning. After that for all rules branched into the sub rules and corresponding confidence for the rules are extimated. After obtaining the confidence for each rule, the decision step is performed. In this step, the confidence of each rule is compared to the best value that means minConf. If the confidence of particular rule is below or less than the bestval then that rule is pruned. Whether the confidence of particular rule is equal to or greater then the bestval then that rule is added to the best Ruleset. Best Ruleset is the rule set which contains the best rule which has higher or equal confident to the bestval i.e., excluding the pruned rules. In the end of the algorithm we are getting the best rules. Thus the rule pruning is processed through the B&B technique.

EXPERIMENTAL RESULTS

For our experimentation, we are taking the Adult dataset which contains information extracted from a 1994 Census made in the USA. A novel process regarding this data set is to predict if an individual's annual income exceeds $50,000.

The original data set contains 45,255 samples with 16 features which includes the 6 continuous and 9 categorical. Preprocessing step is performed such as Filtering and binary expansion. First, the "fnlwgt" (final weight) feature and samples with unknown values were filtered out, which resulted in 30,148 clean samples. Subsequently, categorical features suffered a binary expansion whereas every categorical feature was prolonged in C new binary features, so that the original value is marked as 1, resulting in 106 numerical features. Class was included as a feature for knowledge mining.

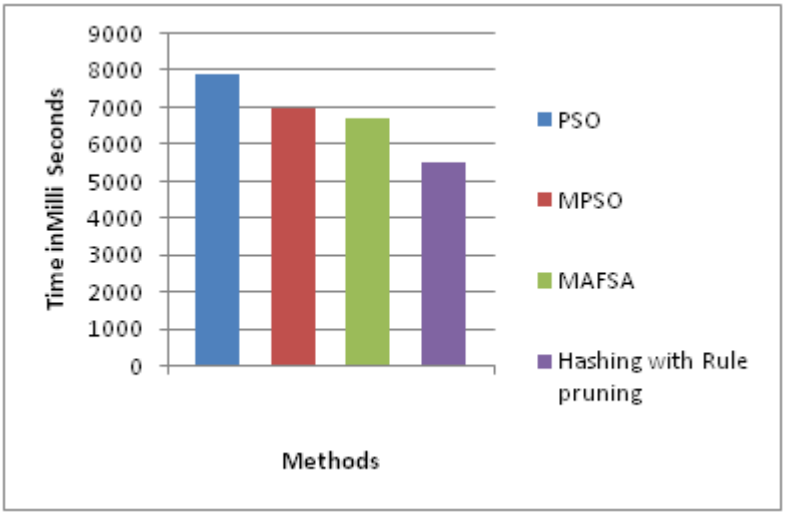

Figure 1: Time Complexity Comparison

In the above graph, we are comparing the proposed hashing with rule pruning with the existing system such as PSO, MPSO and MAFSA in terms of time. In this graph, x axis will be the methods (existing & proposed system) and y axis will be time in ms. The proposed system has lowest time among the other approaches. Thus the proposed system is well effective in time complexity comparatively.

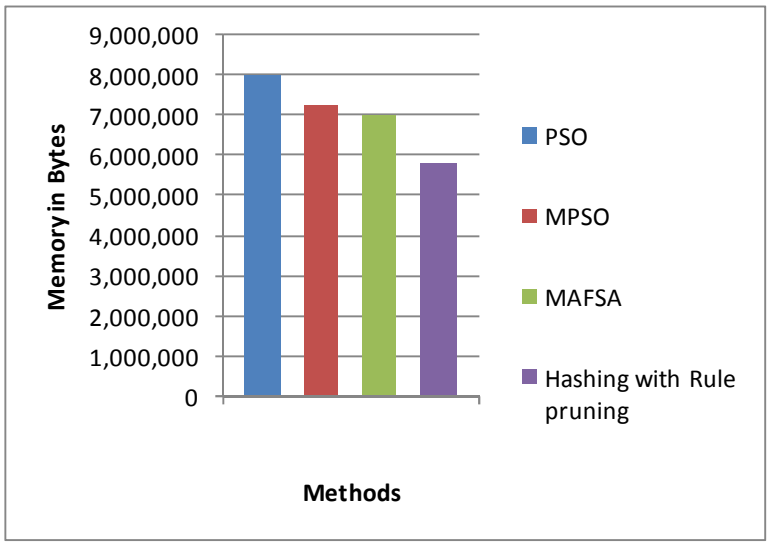

Figure 2: Memory Complexity Comparison

In the above graph, we are comparing the proposed hashing with rule pruning with the existing system such as PSO, MPSO and MAFSA interms of memory. In this graph, x axis will be the methods (existing & proposed system) and y axis will be memory in bytes. The proposed system has lowest memory among the other approaches. Thus the proposed system is well effective in memory or space constraints comparatively.

CONCLUSIONS

Association rule mining is one of the most important techniques in the field of data mining. In this study, we are proposing two novel approaches which are used to enhance the performance of ARM by reducing the memory constraints and increase the accuracy rate of the system. In this study, present a novel dictionary with most terrible case constant lookup time. It has many advantages like simple to implement and high performance comparatively. The majority of the association rules have redundant information and therefore that rules can't able to use for an application directly. So pruning rules is used to obtain very important rules or knowledge. With the intension of this we are proposing a new rule pruning approach based on B&B technique. Our experimental results demonstrate that proposed system is better than existing system and other hash based methods because it efficiently map the item sets in the hash table and it also reduce the space constraints as well as it is increase the accuracy using rule pruning strategy.

REFERENCES

1. H. Toivonen, Sampling large databases for association rules, in: Proceedings of the 22nd VLDB Conference, 1996, pp. 134–145.
2. S. Birn, R. Motwani, J.D. Ullman, S. Tsur, Dynamic itemset counting and implication rules for market basket data, in: Proceedings of the ACM SIGMOD, 1997, pp. 255–264.
3. S. S. Gun, Application of genetic algorithm and weighted itemset for association rule mining, Master Thesis, Department of Industrial Engineering and Management, Yuan-Chi University, 2002.
4. R. J. Kuo, C.W. Shih, Association rule mining through the ant colony system for National Health Insurance Research Database in Taiwan, Computers and Mathematics with Applications 54 (11–12) (2007) 1303–1318.
5. R. J. Kuo, S.Y. Lin, C.W. Shih, Discovering association rules through ant colony system for medical database in Taiwan," to appear, International Journal of Expert Systems with Applications 33 (November (3)) (2007).
6. R. J. Kuoa, C.M. Chaob, Y.T. Chiuc, Application of particle swarm optimization to association rule mining in Applied Soft Computing 11 (2011) 326–336.
7. Bharathi.T and Krishnakumari. P, "An Enhanced Application of Modified PSO for Association Rule Mining ", IJCSI International Journal of Computer Science Issues, Vol. 10, Issue 4, No 1, July 2013.
8. Toivonen, H. Klemetinen, M., Ronkainen, P., Hatonen, K., and Mannila, H. "Pruning and grouping discovered association rules." Mlnet Workshop on Statistics, Machine Learning, and Discovery in Databases, 1995, 47-52.
9. Bayardo, R., Agrawal, R, and Gunopulos, D. "Constraint-based rule mining in large, dense databases." To appear in *ICDE-99*, 1999.
10. Quinlan, R. *C4.5*: *program for machine learning*. Morgan Kaufmann, 1992.
11. Mahta, M., Agrawal, R. and Rissanen, J. "SLIQ: A fast scalable classifier for data mining." EDBT-96.
12. Arnold I. Dumey. Indexing for rapid random access memory systems. Computers and Automation, 5(12):6–9, 1956.

13. Yossi Azar, Andrei Z. Broder, Anna R. Karlin, and Eli Upfal. Balanced allocations. SIAM J. Comput., 29(1):180–200, 1999.

14. J. Lawrence Carter and Mark N. Wegman. Universal classes of hash functions. J. Comput. System Sci., 18(2): 143–154, 1979.

15. Jeanette P. Schmidt and Alan Siegel. On aspects of universality and performance for closed hashing (extended abstract). In Proceedings of the 21st Annual ACM Symposium on Theory of Computing (STOC ’89), pages 355–366. ACM Press, 1989.

16. Jeanette P. Schmidt and Alan Siegel. The analysis of closed hashing under limited randomness (extended abstract). In Proceedings of the 22nd Annual ACM Symposium on Theory of Computing (STOC ’90), pages 224–234. ACM Press, 1990.

17. K. Vanitha and R.Santhi, (2011), “Using Hash Based Apriori Algorithm to Reduce the Candidate 2-itemsets for Mining Association Rule”, Journal of Global Research in Computer Science, Vol.2, No.5, (ISSN-2229-371-X).