A Survey on Improved Algorithms for Mining Association Rules

Hoda Khanali
Department of Industrial Engineering,
Central Tehran Branch, Islamic Azad University,
Tehran, Iran

Babak Vaziri
Department of Industrial Engineering,
Central Tehran Branch, Islamic Azad University,
Tehran, Iran

ABSTRACT

Different types of data, needs of users and variety application problems are lead to produce a range of methods to discover patterns and dependent relationships. This application follows a set of association rules according to know which one of set of objects affects on a set of other objects. This association rules predict the occurrence of an object based on the occurrence of other objects. The associative algorithms have the challenge of redundant association rules and patterns, but studying various methods of association rules is expressive that the recent researches focused on solving the challenges of the tree and lattice structures and their compounds about association algorithms. In this paper, the associative algorithms and their function are described, and finally the new improved association algorithms and the proposed solutions to solve these challenges are explained.

General Terms

Data Mining

Keywords

Frequent item sets, Mining association rules, Data mining.

1. INTRODUCTION

Data mining can be regarded as an algorithmic process that takes data as input and yields patterns such as classification rules, association rules, or summaries as output [1]. Association rule mining is an unsupervised learning method in data mining. As one of the data mining techniques, association rule mining helps find out intriguing relationships among items in a huge database [2]. Therefore, to discovery of the associative relationship many available approaches are that itemset convert to database and data mining operates on that.

Association rule algorithms produce many patterns and rules in a data set. Increasing the number of features leads to produce the large number of rules. All rules not necessarily attractive. Hence, set of rules should be followed which they have most interesting. As a result, it is important that association rules satisfy accepted criteria such as support, confidence, and so on.

- Support is often used to represent the significance of an association pattern. It is also useful from a computational perspective because it has a nice downward closure property that allows us to prune the exponential search space of candidate patterns [3].
- Confidence is often used to measure the accuracy of a given rule. However, it can produce misleading results, especially when the support of the rule consequent is higher than the rule confidence [3].

An association rule with high support and high confidence values may be uninteresting if the confidence of the rule is equal to the marginal frequency of the rule consequent, which means that the antecedent and consequent of the rule are independent. Under these circumstances, the rule would not provide any new information. Besides, if an association rule has a confidence value less than the consequent support, the rule is not of interest [4].

The paper is organized as follows. In Section 2, Association rule algorithms are introduced. Then in Section 3, these algorithms are generally compared. In the next section, comparative analysis of improved approaches discusses. Finally, conclusion and future work are presented in Section 5.

2. ASSOCIATION RULE ALGORITHMS

Efficient and scalable algorithms have been developed for finding frequent itemsets that they lead to the discovery of associations and correlations rules. In general, the algorithms can be viewed as three approaches:

- candidate generation
- pattern-growth
- vertical data format [5]

In this section, the most common algorithms of frequent itemset mining, mining association rules, are described.

2.1 Apriori algorithm

The algorithm [6] is proposed for Boolean association rules. Apriori employs an iterative approach known as a level-wise search, and computes the frequent itemsets in the database through several iterations. Each iteration has two steps: candidate generation and candidate counting and selection [7]. To improve the efficiency of the level-wise generation of frequent itemsets, an important property called the Apriori property , all nonempty subsets of a frequent itemset must also be frequent, is used to reduce the search space [5].

It repeatedly scans the database and checks a large set of candidates by pattern matching, which is especially true for mining long patterns [8]. The scalability is the most important problem of Apriori algorithm, and the complexity of the computation increases exponentially. That is only one of the many factors that influence the development of several new algorithms for association-rule mining [9].

2.2 FP-growth algorithm

FP-growth was first proposed in [10]. The interesting method adopts a divide-and-conquer strategy. First, it compresses the database representing frequent items into a frequent pattern tree, or FP-tree, which retains the itemset association information. It then divides the compressed database into a set of conditional databases, each associated with one frequent item or "pattern fragment" and mines each database separately. For each "pattern fragment" only its associated data sets need to be examined. Therefore, this approach may substantially reduce the size of the data sets to be searched, along with the "growth" of patterns being examined [5]. The FP-tree construction takes exactly two scans of the transaction database: The first scan collects the set of frequent items, and the second scan constructs the FP-tree [8].

The performance gain achieved by the FP-growth is due in most part to the highly compact nature of the FP-tree, which stores only the frequent items in a frequency-descending order, enabling it to maintain as much prefix sharing as possible among the patterns in the transaction database. However, the construction of such an FP-tree requires two database scans, which is the major limitation of using an FP-tree approach for handling a data stream [11].

2.3 Eclat algorithm

The association rule mining algorithm [12] is proposed for mining frequent itemsets using vertical data format. Both the Apriori and FP-growth methods mine frequent patterns from a set of transactions in TID-itemset format, where TID is a transaction ID and itemset is the set of items bought in transaction TID. This is known as the horizontal data format. Alternatively, data can be presented in item-TID format, where item is an item name, and TID-set is the set of transaction identifiers containing the item. This is known as the the vertical data format. Using the Apriori property in the generation of candidate (k+1)-itemset from frequent k-itemsets, another merit of this method is that there is no need to scan the database to find the support of (k+1)-itemset. This is because the TID set of each k-itemset carries the complete information required for counting such support. However, the TID sets can be quite long, taking substantial memory space as well as computation time for intersecting the long sets [5].

3. COMPARISON OF ASSOCIATION RULE ALGORITHMS

According to the above, all algorithms discover the relationships between itemset to association rule mining in databases and data warehouses. In

Table 1 the association rule algorithms are briefly described.

Apriori algorithm is based on the candidate generation, FP-growth algorithm is based on the pattern-growth, and Eclat algorithm is based on the vertical data format so that Apriori algorithm needs n times scanning the database, FP-growth algorithm needs two times scanning the database, and Eclat algorithm does not need to scanning the database. In addition, Eclat algorithm is only recommended in large databases, and has vertical format. On the other hand, unlike two other algorithms Apriori algorithm is not scalable. Except that Eclat algorithm applies a difference set method, three algorithms are not an appropriate memory and time performance. In the following

Table 1, the Features of three algorithms are listed according to the above.

Table 1. Comparison of the association rule algorithms

Algorithm	Apriori	FP-growth	Eclat
Pattern mining method	Candidate generation	Pattern-growth	Vertical data format
The number of scanning the database	n	2	0
For large databases	Not recommended	Not recommended	Recommended
Data format	Horizontal	Horizontal	Vertical
Scalability	No	Yes	Yes
Memory and time performance	No	No	No (Applying a difference set method)
Features	- Generating Search candidates - The structure of the database	- Unique searching way - Using optimization the summary structure - Using the divide and conquer approach	- No need to scan the database

4. IMPROVED ASSOCIATION ALGORITHMS

Predictability-based collective class association rule (PCAR) [13] applies Eclat algorithm structure, and rules evaluation such as rule ranking and pruning a set of non frequent objects uses cross-validation. This algorithm can almost has an accurate prediction due to generate the rules based on a final classifier. In addition, frequent association rules with high confidence, belonging to every class, are classified and class association rules (CARs) are ranked. On the other hand, every association rule with low confidence, not belonging to any class, has little analytical value so it is not considered in the CARs. In this way, evaluated rules are calculated with inner cross-validation at each stage.

A Lattice and Diffset based algorithm for CAR Mining with the Itemset Constraint (LD-CARM-IC) [14] decreases limitations of lattice approach to improve the time and memory usage in mining CARs. The frequent itemsets are maintained by a lattice structure obtained by once scanned the dataset in LD-CARM-IC. Then, the algorithm scrolls the lattice that the production problem of the frequent itemsets is decomposed into several subproblems, and satisfies the constraints. For analyzing each node of the lattice, this algorithm considers the constrained itemset, and to find a path of each child node, it applies the relations of the parent node. Since scrolling the lattice nodes affect on the generated rules, the algorithm marks nodes. Finally, after mining all CARs to satisfy the constraint. In the following, if a new scrolling with the new itemset constraint is needed, all marks of nodes are removed.

A technique [15] applies FP-growth algorithm structure so that a tree named D-tree builds one time scanning the database. The construction of the tree increases the process speed of the manufacturing frequent itemsets. After creating D-tree, the technique records the number of iterations per search as frequency of each item which recorded in a node table. Therefore, the technique considers a new improved FP-tree and the node table as an input, and it finds all association rules with high confidence to join the part of the association rules which sharing on the same prefixes. In this approach, D-tree is the improved FP-tree without the generation of the conditional FP trees.

Novel Equivalence Class Rule tree (NECR-tree) [16] is based on the provided theorems develops mining CARs with class constraints. This algorithm improves an efficient and fast because for each item the support criterion is calculated by a set of object identifiers including both itemset and class, NECR-tree without obtaining the support can find the position of each node, and the constrained itemset not produce. To prune nodes, the method consists of some theorems and lemmas. Therefore, the nodes that not satisfy on the class constraints are discarded. Finally found that NECR-tree not joins the nodes based on the same attributes.

CAR-Miner-Diff-Sort [17] improves version of CAR-Miner [18] such that CAR-Miner builds MECR-tree (a modified data structure of ECR-tree), and then using the subtrees of MECR-tree for each item is characterized different paths. According to the specified paths, CAR-Miner is composed and sorted frequent itemsets based on their corresponding extensions. Despite CAR-Miner computes all intersections between Obidsets (sets of object identifiers that contain itemsets), CAR-Miner-Diff-Sort measures only the difference between two Obidsets (d2O) to reduce the memory is required for storing and time required for calculating the intersection of two Obidsets.

CCAR [19] is based on a tree structure named the Constraint Class Rule tree (CCR-tree). First, the tree includes the constrained itemsets and frequent nodes. Second, only nodes which satisfy the constraint are applied to build the tree. Finally, infrequent itemsets are pruned to increase the running speed of the processes by two theorems. This theorems lead to faster and more effective of this algorithm for mining CARs. Therefore, CCAR to integrating the constraint rules improves the time computing and the storage space.

Minimal non-redundant Multilevel and Cross-level Association Rules (MMCAR) [20] builds the lattices based on CHARM-L [21]. To generate closed itemsets MMCAR scrolls the level-wise, and the itemsets are mined from levels as the cross-levels, the components of the lattices. Since this algorithm has hierarchical levels, CHARM-L is considered to construct the frequent closed itemset lattice (FCIL) for each level. Then, MMCAR applies the features of MG-CHARM [22]. The features lead to determine the minimum number of iterations of the itemsets, and decrease computing time of levels. The algorithm has the cross-level and the multi-level correlation mining benefits which reduce the time required for computing but it has a low analytical about the computing memory.

Multi-Objective Particle swarm optimization algorithm for Association Rules mining (MOPAR) [23] addresses the problem of numerical Association Rules Mining (ARM) by developing a Multi-Objective Particle Swarm Optimization (MOPSO). The algorithm applies three objectives, including confidence, comprehensibility, and interestingness, to improve the trade-off between them. In the first step, three measures to mine numerical ARs, including the confidence, the comprehensibility, and the interestingness are explained. In the next step, MOPAR introduces the particles based on two methods _Pittsburgh and Michigan_ such that the particles are the same chromosomes of the GA algorithms. The Michigan encodes the association rules as particles. In addition, the PSO algorithm measures during the run the developed particles. Finally, by using the Pareto optimality the best association rules are extracted.

MFS_DoubleCons [24] is based on structure of frequent itemsets with double-constraint. This approach presents type of distinguished procedure to generate the frequent itemsets. So after, MFS_DoubleCons considers the mining of frequent itemsets. Monotone and anti-monotone constraints are used to increase the mining efficiency and reduce the search space in this algorithm. Having discovered frequent itemsets, MFS_DoubleCons introduces a structure of frequent itemsets. The efficient method attempts without further reference to the database and creating the lattice of closed itemset to reduce the number of patterns extracted.

STreeDC-Miner and STreeNDC-Miner [25] obtains frequent similar patterns based on STree structure and the similarity function which has the f-downward closure property. Using the f-downward closure property, the algorithms finds frequent similar pattern, and using an adaptation of the GenRules algorithm [26] the algorithms generates association rules from patterns. Both algorithms apply similarities different from the equality instead the equality, and they improve and the number of frequent frequent similar pattern and the computation time.

Highly Coherent Association-Rule Mining (HCARM) [27] is Apriori-like approach, namely based on the mining of positive association rules, for transaction databases. According to provide a coherent rules, the algorithm introduces four conditions which focus on minimum support and minimum confidence. After finding the highly coherent association rules from transactions, lower and upper bounds of itemsets are derived. The bounds of itemsets help to discard non frequent itemsets which are not included highly coherent itemsets. Contingency tables meanwhile explore the logical conditions. Therefore, HCARM reduce the time to generate the itemsets, and accelerates the mining process.

CAR-Miner [18] proposes MECR-tree structure, a modified structure of the equivalence class rule-tree (ECR-tree) [28], which derives all frequent itemsets. For this reason, the database size decreases, and the computing speed of candidate rule increases. Then, the itemsets are classified based on recognized similarities, the same attributes, as one node is accordingly considered in the tree. However, the algorithm focuses on some theorems pruning nodes, can quickly compute the support of itemsets, and in the tree not scans many nodes. CAR-Miner improves the performance but it not reviews about time and memory.

5. COMPARISON OF IMPROVED ASSOCIATION ALGORITHMS

Due to the above subjects, the various improvements of the available algorithms on discovery of association rules are provided, and new algorithms are introduced based on the initial algorithms or the combination of these algorithms with other methods. Each of these algorithms has proposed the solutions to reduce the impact of the challenges of the association rules briefly mentioned in

Table 2.

Table 2. Evaluation of improved association algorithms

Algorithms	**Structures**	**Discussed challenges**	**Solutions**
PCAR [13]	- Eclat algorithm	- Existing many redundant rules	- Applying cross-validation and aggregating the resulting rules
LD-CARM-IC [14]	- Lattice structure	- Existing many redundant association rules	- Usage of a lattice structure for mining CARs and a memory reduction strategy
The algorithm [15]	- FP-growth algorithm	- The complex inter mediate process of the frequent item set generation	- Improve FP-tree and a frequent item set mining algorithm
NECR-tree [16]	- NECR-tree structure	- Existing redundant or unimportant rules	- Mining relevant CARs that considers constraints on the rule consequent
CAR-Miner-Diff-Sort [17]	- Tree structure	- memory consumption and run time of CAR-Miner	- Useing the difference between two Obidsets (d2O) to save memory usage and run time
CCAR [19]	- CCR-tree structure	- Mining CARs with itemset constraints	- The Constraint Class Rule tree (CCR-tree) structure
MMCAR [20]	- lattice and hierarchical structure	- Mining minimal multilevel and cross-level association rules	- Generating hierarchical minimal rules
MOPAR [23]	- Evolutionary Algorithms (EAs) and multi-objective particle swarm optimization (PSO) algorithm	- Discovering numerical association rules (ARs)	- Use rough values containing lower and upper bounds
MFS_DoubleCons [24]	- Structure of frequent itemsets with double-constraint	- Mining frequent itemsets	- Mining frequent itemsets with double-constraint
STreeDC-Miner and STreeNDC-Miner [25]	- STree structure - Mixed data using Boolean similarity functions	- Mining frequent patterns and association rules	- Pruning the search space of frequent similar patterns
HCARM [27]	- Apriori algorithm	- Common sense rules and misleading rules	- Applying logic equivalence for coherent rules
CAR-Miner [18]	- MECR-tree structure	- Accuracy classifier	- Design the MECR-tree structure

According to

Table 2 PCAR algorithm applies Eclat algorithm structure. LD- CARM-IC and MMCAR algorithms based on lattice structure. The algorithm presented at [15] applies FP growth algorithm structure. NECR-tree, CAR-Miner-Diff-Sort, CCAR, CAR-Miner, and the algorithm presented at [25] based on tree structure. MOPAR algorithm applies Evolutionary Algorithms (EAs) and multi-objective particle swarm optimization (PSO) algorithm. MFS_DoubleCons applies structure of frequent itemsets with double-constraint. HCARM algorithm applies Apriori algorithm structure.

Here, it should be noted that PCAR algorithm through cross-validation and aggregating the resulting rules, LD- CARM-IC algorithm through a lattice structure for mining CARs and a memory reduction strategy, and NECR-tree algorithm through mining relevant CARs that considers constraints on the rule consequent solve many redundant rules.

the algorithm presented at [15] solves the complex inter mediate process of the frequent item set generation through FP-tree and a frequent item set mining algorithm, CAR-Miner-Diff-Sort algorithm solves memory consumption and run time of CAR-Miner through the difference between two Obidsets (d2O), CCAR algorithm solves itemset constraints through The Constraint Class Rule tree (CCR-tree) structure, MMCAR algorithm solves mining minimal multilevel and cross-level association rules through hierarchical minimal rules, MOPAR algorithm solves numerical association rules (ARs) through lower and upper bounds, MFS_DoubleCons algorithm solves mining frequent itemsets through double-constraint, the algorithm presented at [25] solves Mining frequent patterns and association rules through pruning the search space of frequent similar patterns, HCARM algorithm solves common sense rules and misleading rules through logic equivalence, CAR-Miner algorithm solve accuracy classifier through MECR-tree structure.

6. CONCLUSION

In this pepar, the primary concepts and techniques which detecting repeating patterns of common algorithms of mining association rules are analyzed. First, a brief description is presented about association rule algorithms, and a general comparison of these algorithms is discussed. Then, recent related works are introduced. It is understood by analyzing each of this works that the volume of data is growing nowadays, and in recent years the researches are to achieve purposes such as performance, scalability and memory usage improvement. In this field, many issues also need to more study. Therefore, improving the class association rules with tree structures, lattice structures and methods of pruning the pattern and data space could be an interesting issue for future research.

7. REFERENCES

[1] Geng, L., and Hamilton, H. 2006. Interestingness measures for data mining: A survey. ACM Computing Surveys (CSUR).

[2] Kim, C., Lee, H., Seol, H., and Lee, C. 2011. Identifying core technologies based on technological cross-impacts: An association rule mining (ARM) and analytic network process (ANP) approach. Expert Systems with Applications.

[3] Tan, P., Kumar, V., and Srivastava, J. 2004. Selecting the right objective measure for association analysis. Information Systems.

[4] Luna, J., Romero, J., and Ventura, S. 2013. Grammar-based multi-objective algorithms for mining association rules. Data & Knowledge Engineering.

[5] Han, J., Kamber, M., and Pei, J. 2011. Data mining: concepts and techniques. Elsevier.

[6] Agrawal, R., and Srikant, R. 1994. Fast algorithms for mining association rules. in 20th International Conference on Very Large Data Bases.

[7] Cao, L. 2009. Data mining and multi-agent integration. Springer Science & Business Media.

[8] Han, J., Pei, J., Yin, Y. and Mao, R. 2004. Mining frequent patterns without candidate generation: A frequent-pattern tree approach. Data mining and knowledge discovery.

[9] Kantardzic, M., 2011. Data mining: concepts, models, methods, and algorithms. John Wiley & Sons.

[10] Han, J., Pei, J., and Yin, Y. 2000. Mining frequent patterns without candidate generation. ACM Sigmod Record.

[11] Tanbeer, S., Ahmed, C., Jeong, B., Lee, Y., and Sliding, w. 2009. Sliding window-based frequent pattern mining over data streams. Information sciences.

[12] Zaki, M., Parthasarathy, S., Ogihara, M., and Li, W. 1997. New algorithms for fast discovery of association rules. in 3rd International Conference on Knowledge Discovery and Data Mining (KDD'97).

[13] Song, K., and Lee, K. 2017. Predictability-based collective class association rule mining. Expert Systems with Applications.

[14] Nguyen, D., Nguyen, L., Vo, B., and Pedrycz, W. 2016. Efficient mining of class association rules with the itemset constraint. Knowledge-Based Systems.

[15] Narvekar, M., and Syed, S. 2015. An Optimized Algorithm for Association Rule Mining Using FP Tree. Procedia Computer Science.

[16] Nguyen, D., Nguyen, L., Vo, B., and Hong, T. 2015. A novel method for constrained class association rule mining. Information Sciences.

[17] Nguyen, L., and Nguyen, N. 2015. An improved algorithm for mining class association rules using the difference of Obidsets. Expert Systems with Applications.

[18] Nguyen, L., Vo, B., Hong, T., and Thanh, H. 2013. CAR-Miner: An efficient algorithm for mining class-association rules. Expert Systems with Applications.

[19] Nguyen, D., Vo, B., and Le, B. 2015. CCAR: An efficient method for mining class association rules with itemset constraints. Engineering Applications of Artificial Intelligence.

[20] Hashem, T., Ahmed, C., Samiullah, M., Akther, S., and Jeong, B. 2014. An efficient approach for mining cross-

level closed itemsets and minimal association rules using closed itemset lattices. Expert Systems with Applications.

[21] Zaki, M., and Hsiao, C. 2005. Efficient algorithms for mining closed itemsets and their lattice structure. knowledge and data engineering.

[22] Vo, B., and Le, B. 2009. Fast algorithm for mining minimal generators of frequent closed itemsets and their applications. in Computers & Industrial Engineering.

[23] Beiranvand, V., Mobasher-Kashani, M., and Bakar, A. 2014. Multi-objective PSO algorithm for mining numerical association rules without a priori discretization. Expert Systems with Applications..

[24] Duong, H., Truong, T., and Vo, B. 2014. An efficient method for mining frequent itemsets with double constraints. Engineering Applications of Artificial Intelligence.

[25] Rodríguez-González, A., Martínez-Trinidad, J., Carrasco-Ochoa, J., and Ruiz-Shulcloper, J. 2013. Mining frequent patterns and association rules using similarities. Expert Systems with Applications.

[26] Agrawal, R., and Srikant, R. 1994. Fast algorithms for mining association rules. in 20th int. conf. very large data bases, VLD.

[27] Chen, C., Lan, G., Hong, T., and Lin, Y. 2013. Mining high coherent association rules with consideration of support measure. Expert Systems with Applications.

[28] Vo, B. and Le, B. 2008. A novel classification algorithm based on association rules mining. in InPacific Rim Knowledge Acquisition Workshop.

IJCA™ : www.ijcaonline.org