I J C T A, 9(34) 2016, pp. 459-465
© International Science Press

Mining Association Rules from Large Volumes of Data : A Survey

V. Ramya[1] and M. Ramakrishnan[2]

ABSTRACT

Certainly association rule mining has become an important tool for market researchers to analyze the historical transaction data and extract useful rules. Association rule mining is having applications in varieties of fields and market basket analysis is one among them. The generated strong rules are used by market experts to plan layouts and place items accordingly in a close proximity so that revenue generation can be increased. In this paper, a literature survey on various association rule mining methods are discussed. Few algorithms are dependent on generating frequent itemsets, few use tree structure to represent the items and few generate frequent itemsets without generating candidate itemsets. FP-growth and RARM uses tree structures to represent the data and its advantages are that large volumes of database transactions can be expressed in a compact way. Elcat algorithm uses support matrix to generate rules. All the algorithm are survey under different perspectives viz. number of scans over the database, whether they generate candidate itemset or not, and the data structure used to store the transactions. The paper also presents the comparative statement of the methods discussed along with their advantages and disadvantages.

Keywords: Association Rule Mining, ARM, Frequent Itemsets, Apriori, Large Data

1. INTRODUCTION

Recent changes in data collection and storage technologies have given liberty to many organizations to keep vast amount of data relating to their business activities. The purpose of storing huge volume of transactional data is that a new knowledge or idea can be extracted from them[1]. Data mining is the process used to discover useful knowledge from large volumes of data. This knowledge can be patterns, associations, changes, anomalies, significant structures, etc. Data mining is not a single technique rather than collection of various methods and procedures, each one is having significance in knowledge discovery process and each method analyses the data under different perspectives[2].

Various methods of data mining include association, classification, clustering, similarity analysis, summarization, sequential pattern discovery, prediction, etc. The important and most popular pattern discovery method in data mining is association rule mining, which was introduced by Agarwal et al [3] in the year 1993. It tries to find the interesting correlations, associations, casual structures and frequent patterns among set of items in the dataset. The associations between data items are expressed in the form of association rules and the process of finding such rules is termed as association rule mining.

The real motivation for association rule mining arouse from the need to analyze the supermarket transaction data to examine the customer purchase behaviour of various products. Termed as Market Basket Analysis, it tries to find the combination of items that appear together in several transactions of a customer. It finds the frequent itemsets using user specific minimum support[4].

Using frequent itemsets, strong association rules are derived and these rules are filtered using user defined minimum confidence. Generally most of the association rule mining algorithm works on frequent

[1] Research Scholar, Department of Computer Science, Bharathiyar University, Coimbatore, Tamil Nadu, India, *E-mail: vramya11@gmail.com*

[2] Chairperson, School of Information Technology, Madurai Kamaraj University, Madurai, Tamil Nadu, India, *E-mail: ramkrishod@gmail.com*

itemset mining. These algorithms find rules which have high support and high confidence[5]. Infrequent itemset mining has not explored that much in research area.

A typical and widely used application of association rule mining is market basket analysis. Other applications of association rule mining includes medical diagnosis, CRM of credit card business, protein sequences, etc[6]. In medical diagnosis, association rules are used to assist physicians to cure patients. The problem of reliable diagnostic rules is hard because theoretically no induction process can correct induced hypothesis.

Association rule mining is used as tool for rapid determination of DNA sequences and by inference, the amino acid sequences of proteins from structural genes. Customer relationship management is used to identify the customer groups, products and services. Association rule mining allows credit card marketing personnel to know their customers behaviour well to provide better services. Since association rule mining is an important activity in data mining process, this paper presents a survey on various association rule mining methods. The paper has been structured as follows: Introduction to data mining and association rule mining techniques are discussed in section 1. Basics of association rule mining problem is discussed in section 2. Section 3 present detail explanations about various association rule mining techniques. Merits and demerits of various rules are discussed in section 4 and paper ends by presenting conclusion in section 5.

2. BACKGROUND

The formal statement of association rule mining problem is as follows: Let $I = I_1, I_2,\ldots I_m$ be a set of m distinct attributes, T be transaction that contains a set of items such that $T \subset I$, let D be the dataset with different transactional records.

An association rule is an if-then-else form of representation such as $X \Rightarrow Y$ where $X,Y \in I$ are set of items called itemsets and $X \cap Y \neq 0$. X is said to be antecedent and Y is said to be consequent.

The two important metrics for association rules are support and confidence[7]. As database contains huge number of records and user concern is only about frequently purchased items, many association rules are generated. In order to filter out the weak and unwanted rules, support and confidence thresholds are used. These two are the minimum constraints imposed by the algorithm and additional constraints can be imposed by the user.

Support is the percentage of records in the dataset that contain X È Y with the total number of records in the dataset D[8]. Count for an item is increased by one every time the item is encountered in various transactions T. Support does not exhibit the quantity of the item in to account. The formula to calculate support is as follows

$$support\,(XY) = \frac{Support\ of\ XY}{Total\ no.of\ transactions\ in\ D}$$

Support of an item represents the statistical significance of an association rule. Before performing association rule mining, users specify the minimum support and rules whose support is below the minimum support are not included in the association rule list. Sometimes infrequent itemsets that are below the minimum support are still important in association rule mining process.

Confidence is defined as the percentage of transactions that contain $X \cup Y$ with the total number of records that contain X. Confidence is calculated as follows:

$$confidence\,(X \mid Y) = \frac{Support\,(XY)}{Support\,(X)}$$

All the association rules are objective and uses statistical measure to determine which rules are useful. Using confidence, the unimportant rules are removed and useful ones are considered as knowledge.

3. ASSOCIATION RULE MINING APPROACHES

Many algorithms have been proposed for generating association rules. Most of the algorithms work by generating frequent itemsets. Generally association rule mining is a two step process. In the first step, frequent itemsets are generated using the support specified by the user. From the frequent itemsets, association rules are generated and by using confidence level, strong rules are selected. In this section, we will discuss important association rule mining techniques.

3.1. AIS Algorithm

Agarwal, Imielinski and Swami algorithm, abbreviated as AIS, was the first proposed algorithm for association rule mining[9]. The primary focus of AIS algorithm is to improve the quality of the dataset with support to decision queries. This algorithm generates only one itemset consequent rule, containing one item. AIS algorithm scans the database many times to generate frequent itemsets.

AIS algorithm works in three phases. In the first phase, candidate itemsets are generated and counted on the fly as the database is scanned. In the second phase, it determines which of the large itemsets of the previous pass are available in the transaction. In the third phase, new candidate itemsets are generated by extending these large itemsets with other items of a transaction.

An estimation method was introduced to prune those itemsets that cannot be enlarged. This improves the performance of AIS algorithm and reduces the unnecessary effort of counting itemsets[10]. In AIS algorithm, all the candidate itemsets and frequent itemsets are stored in main memory and hence AIS needs memory management[11]. Another enhancement of AIS is to delete candidate itemsets that have never been extended.

3.2. SETM Algorithm

This algorithm also does on-the-fly counting like AIS. SETM was created to support SQL and relational database operations[12]. SETM uses standard SQL join operation to generate candidate itemsets. It separates candidate generation from counting. During each iteration, the support count of candidate itemsets is determined by aggregating sequential structures[12].

SETM works in three phases. Like AIS, candidate itemsets are generated on-the-fly with multiple database scans, but counted at the end of pass. In the second phase, new candidate items are generated but the TID of the generating transaction is stored in a sequential structure along with candidate itemset. In the third phase, support count of candidate itemsets are determined by aggregating sequential structure.

3.3. Apriori Algorithm

One of the commonly used algorithms for association rule mining is Apriori, which works by finding frequent itemsets. Apriori is more efficient in candidate generation process because Apriori uses different candidate generation method and usage of pruning technique[13]. There are two steps to find the large itemsets using Apriori algorithm. First candidate itemsets are generated, followed by scanning database to find support count of corresponding itemsets. The itemsets that does not have minimum support are pruned in the next step.

In each step, only candidate itemsets that have minimum support are alone generated and checked. While finding frequent itemsets, Apriori avoids effort wastage in counting itemsets that are infrequent. Next level candidate itemsets are generated by joining the frequent itemsets of previous level in a wise manner. Apriori algorithm dramatically reduces the I/O cost and memory requirements[14].

3.4. Apriori TID Algorithm

A variation of Apriori algorithm, called Apriori TID, uses generation function in order to determine the candidate itemsets[15]. The difference between Apriori and Apriori TID is that in later database is not referred for counting support after first pass itself. If a transaction does not have candidate k-itemset, then that candidate itemset should not have any entry for that transaction. This reduces the number of transaction in the set containing the candidate itemsets[15].

Here the database is not used for counting the support of candidate itemsets after the first pass. Candidate itemsets are generated the same way as in Apriori algorithm. A new set is generated in which each member has TID of each transaction and large itemsets are present in a transaction. This set is used for counting support of each candidate itemset.

3.5. Apriori Hybrid Algorithm

This is another variation of traditional Apriori algorithm. It is the combination of traditional Apriori and Apriori TID algorithms. Apriori hybrid algorithm uses traditional version of Apriori in the initial pass and switches over to Apriori TID when it expects the candidate itemset[16]. Even though switching of algorithm takes time, still it is better in most of the cases.

3.6. FP – Growth Algorithm

It uses tree structure to find frequent itemsets. Frequent pattern mining breaks the bottleneck of Apriori and frequent itemsets are generated with only two passes over the database. Frequent itemsets are generated without candidate generation process. Since it avoids candidate generation process and minimum pass over the database, FP-Growth algorithm is faster than Apriori[17].

Frequent patterns are generated in two steps, FP tree construction and getting frequent patterns. FP growth algorithm adopts divide and conquer strategy[18]. Algorithm retains itemset associated information and compressed databases are divided into set of conditional databases, and each one is associated with frequent itemset. FP-Growth algorithm needs minimum memory for storing transactions[18].

In FP-Growth tree, each node represents one item and each path represents set of transactions that involve with particular item[19]. All the transactions that contain the same item can be traced and counted by following the link. This makes large databases to be compressed into small FP tree structure.

For three reasons FP–Growth algorithm is better. First, FP tree is a compressed representation of original database and irrelevant information are pruned. FP tree algorithm scans the database only twice, one for FP-tree construction and other for mining frequent patterns. Since this algorithm uses divide and conquer method, it considerably reduces the size of subsequent conditional FP-tree.

3.7. Rapid Association Rule Mining

RARM also uses tree structure to represent the original database and avoids candidate generation. RARM uses SOTrieIT (support Ordered Trie Itemset) structure and generates 1-itemset, 2-itemset quickly without scanning the database[20]. Like FP-tree, every node of SOTrieIT represents one item and relevant support count.

It follows the same procedure as FP-tree to built TrieIT. For each transaction, all possible itemset combinations are extracted and items that are already included in the TrieIT, its support count is increased by 1. If an item is not found in TrieIT, then that item is inserted as new node.

To construct SOTrieIT, only 1-itemsts and 2-itemsets are extracted for each transaction. Building process is same like FP-tree. To mine large itemsets, SOTrieIT tree is traversed in a depth-first search fashion[20].

It starts from left most first level node, and continues to generate 1-itemset. 2-itemsets are also generated easily and SOTrieIT is much faster than FP-tree.

3.8. ELCAT Algorithm

It represents the transaction as a bit matrix and intersection of rows gives the support count of itemsets[21]. It also follows depth first search traversal for finding frequent itemsets. Bit matrices are used to represent transactions in which each row corresponds to an item and each column represents a transaction[21]. Value 1 is fixed if the item corresponding to the row is contained in corresponding transaction column, otherwise it is cleared. Fixing value 1 can be replaced as fixing true and false value. Rows corresponding to infrequent itemsets are discarded from the constructed matrix. This can be done conveniently by inserting item identifier for each row.

In the first scan, a TID list is maintained for each item. Next itemset is generated from previous one using Apriori property and depth first search computation. This process is continued until no candidate itemset can be left. ELCAT algorithm avoids the overhead of generating all the subsets of a transaction and checking them against the candidate has tree during support counting[22].

4. COMPARISION OF VARIOUS OF ARM METHODS

Method	*Advantages / Disadvantages*
AIS	**Advantages** ▪ First algorithm to introduce association rule mining problem **Disadvantages** ▪ Makes multiple passes over the database. ▪ Generates and counts too many candidate itemsets that turn out to be small, and needs more space
SETM	**Advantages** ▪ Designed to support SQL and RDBMS **Disadvantages** ▪ Like AIS, it makes multiple scans over the database ▪ For each candidate itemset, there are many entries as it support value
Apriori	**Advantages** ▪ Any subset of a frequent itemset is also a frequent itemset. ▪ This reduces the number of candidates being considered by only exploring the itemsets whose count is greater than minimum support count **Disadvantages** ▪ Still inherits the drawback of scanning whole database many times
Apriori TID	**Advantages** ▪ Number of entries in 'C' set may be smaller than the number of transactions in the database **Disadvantages** ▪ Still dependent on Apriori algorithm
Apriori Hybrid	**Advantages** ▪ Even though switching of algorithm takes time, still it is better in most of the cases. **Disadvantages** ▪ Extra cost is incurred while shifting from Apriori to Apriori TID
FP-Growth	**Advantages** ▪ It represents big databases into compact tree structure ▪ Algorithm needs only two scans over the database.

contd. table

Method	*Advantages / Disadvantages*
	Disadvantages
	▪ Difficult to be used in an interactive mining system
	▪ Not suitable for incremental mining.
RARM	**Advantages**
	▪ Better performance because of SOTrieIT
	▪ Generating 1-itemset, 2-itemset can easily be extracted using SOTrieIT
	Disadvantages
	▪ Difficult to be used in an interactive mining system
	▪ Not suitable for incremental mining.
ELCAT	**Advantages**
	▪ To count the support of k+1 large itemsets, there is no need to scan the database
	▪ Algorithm avoids overhead of generating all subsets of a transaction
	Disadvantages
	▪ It requires the virtual memory to perform the transformation

5. CONCLUSION

In this paper, a survey on various methods of association rule mining is discussed in an elaborate manner. Indeed, association rule mining is an important concept in data mining process, which provides correlations among different items of a transaction. We made the survey considering the important algorithms. The algorithms include AIS, SETM, Apriori, FP-growth, RARM and Elcat. The variations of Apriori algorithm viz. Apriori TID and Apriori Hybrid are also discussed. All the findings have been tabulated in section 4 as the comparative study. The table shows that majority of the algorithm is dependent on generating candidate itemsets for generating association rules. Few algorithms use tree structure to represent the big databases in a compact way. From the table it is evident that FP-growth algorithm outperforms other ARM methods.

REFERENCES

[1] Han, Jiawei, Micheline Kamber, and Jian Pei. Data mining: concepts and techniques. Elsevier, 2011.

[2] Witten, Ian H., and Eibe Frank. Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann, 2005.

[3] Agrawal, Rakesh, and Ramakrishnan Srikant. “Fast algorithms for mining association rules.” Proc. 20th int. conf. very large data bases, VLDB. Vol. 1215. 1994.

[4] Menon, Rakesh, et al. “The needs and benefits of applying textual data mining within the product development process.” Quality and reliability engineering international 20.1 (2004): 1-15.

[5] Kotsiantis, Sotiris, and Dimitris Kanellopoulos. “Association rules mining: A recent overview.” GESTS International Transactions on Computer Science and Engineering 32.1 (2006): 71-82.

[6] Rajak, Akash, and Mahendra Kumar Gupta. “Association rule mining-applications in various areas.” Proceedings of International Conference on Data Management, Ghaziabad, India. 2008.

[7] Liu, Duen-Ren, and Ya-Yueh Shih. “Integrating AHP and data mining for product recommendation based on customer lifetime value.” Information & Management 42.3 (2005): 387-400.

[8] Chui, Chun-Kit, Ben Kao, and Edward Hung. “Mining frequent itemsets from uncertain data.” Advances in knowledge discovery and data mining. Springer Berlin Heidelberg, 2007. 47-58.

[9] Tsai, Pauray SM, and Chien-Ming Chen. “Mining interesting association rules from customer databases and transaction databases.” Information Systems 29.8 (2004): 685-696.

[10] Chang, Joong Hyuk, and Won Suk Lee. “Finding recent frequent itemsets adaptively over online data streams.” Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2003.

[11] Orlando, Salvatore, *et al.* “Adaptive and resource-aware mining of frequent sets.” Data Mining, 2002. ICDM 2003. Proceedings. 2002 IEEE International Conference on. IEEE, 2002.

[12] Rao, Chinta Someswara, et al. "Mining Association Rules Based on Boolean Algorithm-a Study in Large Databases." International Journal of Machine Learning and Computing 3.4 (2013): 347.

[13] Dong, Jie, and Min Han. "BitTableFI: An efficient mining frequent itemsets algorithm." Knowledge-Based Systems 20.4 (2007): 329-335.

[14] Han, Jiawei, et al. "Mining frequent patterns without candidate generation: A frequent-pattern tree approach." Data mining and knowledge discovery 8.1 (2004): 53-87.

[15] Li, Zhi-Chao, Pi-Lian He, and Ming Lei. "A high efficient AprioriTid algorithm for mining association rule." Machine Learning and Cybernetics, 2005. Proceedings of 2005 International Conference on. Vol. 3. IEEE, 2005.

[16] Dunham, Margaret H., *et al.* "A survey of association rules." Retrieved January 5 (2001): 2008.

[17] Song, Wei, Bingru Yang, and Zhangyan Xu. "Index-BitTableFI: An improved algorithm for mining frequent itemsets." Knowledge-Based Systems 21.6 (2008): 507-513.

[18] Said, Aiman Moyaid, P. D. D. Dominic, and Azween B. Abdullah. "A comparative study of fp-growth variations." International Journal of Computer Science and Network Security 9.5 (2009): 266-272.

[19] Li, Haoyuan, *et al.* "Pfp: parallel fp-growth for query recommendation." Proceedings of the 2008 ACM conference on Recommender systems. ACM, 2008.

[20] Woon, Yew-Kwong, Wee-keong Ng, and Ee-Peng Lim. "A support-ordered trie for fast frequent itemset discovery." Knowledge and Data Engineering, IEEE Transactions on 16.7 (2004): 875-879.

[21] Ramaraj, E., and N. Venkatesan. "AN EFFICIENT PATTERN MINING ANALYSIS IN HEALTH CARE DATABASE." Journal of Theoretical & Applied Information Technology 5.5 (2009).

[22] Nath, B., D. K. Bhattacharyya, and A. Ghosh. "Incremental association rule mining: a survey." Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 3.3 (2013): 157-169.