International Journal of Environmental Engineering Science and Technology Research
Vol. 3, No. 1, June 2015, pp. 1 - 8, ISSN: 2326 - 3113 (Online)
Available online at http://ijestr.org/

Review article

Mining of Association Rules: A Review Paper

Miss. Pooja Rajendra Harne

Department of PG Studies
Prof Ram Meghe College of Engineering & Management
Badnera-Amravati, India
Phone No- 9730038317
Email-address pooja.harne91@gmail.com

S.D. Deshpande

Department of PG Studies
Prof Ram Meghe College of Engineering & Management
Badnera-Amravati, India
Phone No- 9422926419
Email-address sddeshpande13@gmail.com

Abstract

Frequent pattern mining is one of the active research themes in data mining. It plays an important role in all data mining tasks such as clustering, classification, prediction, and association analysis. Identifying all frequent patterns is the most time consuming process due to a massive number of patterns generated. In this paper, we present thus study of different data structures used and the algorithms for finding the frequent items in the fastest way. We also present an approach for mining of association rule.

Keywords: Data mining; Frequent itemsets; Distributed computation, Association rules

Introduction

Data mining is a powerful new technology with great potential to help companies focus on the most important information in the data they have collected about the behavior of their customers and potential customers. Data mining, the extraction of hidden predictive information from large databases, is a powerful new technology with

great potential to help companies focus on the most important information in their data warehouses. Data mining tools predict future trends and behaviors, allowing businesses to make proactive, knowledge-driven decisions. The automated, prospective analyses offered by data mining move beyond the analyses of past events provided by retrospective tools typical of decision support systems.

Data mining is the process of analyzing data from different perspectives and summarizing it into useful information. Data mining software is one of a number of analytical tools for analyzing data. Data mining sometimes called data or knowledge discovery. Data mining, also known as knowledge discovery in databases, has been recognized as a new area for database research. The area can be defined as efficiently discovering interesting rules from large collections of data. Technically, data mining is the process of finding correlations or patterns among dozens of fields in large relational databases. Correlation typically is association rules.

Association Rule –

Association Rule mining is one of the most important data mining tools used in many real life applications [1][4][5][6][7]. In data mining, association rule learning is a popular and well researched method for discovering interesting relations between variables in large databases. An association rule has two parts, an antecedent (if) and a consequent (then). An antecedent is an item found in the data. A consequent is an item that is found in combination with the antecedent.

Association rules are created by analyzing data for frequent if/then patterns and using the criteria support and confidence to identify the most important relationships. Support is an indication of how frequently the items appear in the database. Confidence indicates the number of times the if/then statements have been found to be true. Finding this kind of association rules, frequent itemsets are required.

The organization of this paper is as follows. Next Section discusses the exact problem statement. This is followed by the literature review of different data structures used and algorithms used. This section is followed by mining the association rules from frequent itemsets. Lastly, we propose an approach for mining of association rules where the data is large and distributed.

I. Problem Statement

Association Rule mining is one of the most important data mining tools used in many real life applications[4],[5]. In this paper, we will discuss the problem of computing association rules within a horizontally partitioned database. We assume homogeneous databases. To mine the association rules the first task is to generate the frequent itemsets. Second task is to mine the association rules from the frequent itemsets.

a. Generation of Frequent Itemsets

Frequent item sets from different databases come to a global database [1],[5]. Since there are so many databases through which frequent data goes to the global database so this increases the number of messages that need to be

passed so as to find frequent k item set. The major problem with frequent set mining methods is the explosion of the number of results and so it is difficult to find the most interesting frequent item sets. So the concept of Finding frequent itemsets from the database which is at Different Distributions, and mine the association rules has been highlighted in this paper.

Mining associations Rules

Following the original definition by Agrawal the problem of association rule mining is defined as[5] : Let I = {i1, i2, ..., in} be a set of n binary attributes called items. Let D = {t1, t2, .., tn} be a set of transactions called the database. Each transaction in D has a unique transaction ID and contains a subset of the items in I. A association rule is defined as an implication of the form X->Y where X, Y ⊆ I and $X \cap Y\_ = \emptyset$. The sets of items (for short itemsets) X and Y are called antecedent (left-hand-side or LHS) and consequent (right-hand-side or RHS) of the rule respectively. An example of an association rule would be "If a customer buys a bread, he is 80% likely to also purchase milk."

Given a minimum confidence threshold minconf and a minimum support threshold minsup, the problem is to generate all association rules that have support and confidence greater than the user-specified minimum support and minimum confidence . In the first pass, the support of each individual item is counted, and the large ones are determined. In each subsequent pass, the large itemsets determined in the previous pass is used to generate new itemsets called candidate itemsets. The support of each candidate itemset is counted, and the large ones are determined. This process continues until no new large itemsets are found.

II. Review of Different Data Structures Used

Trie Data Structure

A trie is a tree data structure that allows strings with similar character prefixes to use the same prefix data and store only the tails as separate data. One character of the string is stored at each level of the tree, with the first character of the string stored at the root. The term trie comes from "re**trie**val**" .** Due to this etymology it is pronounced [tri] ("tree"), although some encourage the use of "try" in order to distinguish it from the more general tree.This trie data structure is used for storing frequent itemsets.

III. Review of Different Algorithms Used

1. AIS Algorithm

The AIS algorithm [17] was the first algorithm proposed by Agrawal , Imielinski , and Swami for mining association rule. It focuses on improving the quality of databases together with necessary functionality to process decision support queries. In this algorithm only one item consequent association rules are generated, which means that the consequent of those rules only contain one item, for example, rules like $X \cap Y \Rightarrow Z$ can be generated but not the rules like $X \Rightarrow Y \cap Z$.

AIS algorithm consists of two phases. The first phase constitutes the generation of the frequent itemsets. This is followed by the generation of the confident and frequent association rules in the second phase. The drawback of the AIS algorithm is that it makes multiple passes over the database. Furthermore ,it generate and counts too many candidate itemsets that turn out to be small, which requires more space and waste much efforts that turned out to be useless.

2. SETM Algorithm

In the SETM algorithm, candidate itemsets [16] are generated on-the-fly as the database is scanned, but counted at the end of the pass. Then new candidate itemsets are generated the same way as in AIS algorithm, but the transaction identifier TID of the generating transaction is saved with the candidate itemset in a sequential structure. It separates candidate generation process from counting. At the end of the pass, the support count of candidate itemsets is determined by aggregating the sequential structure. The SETM algorithm [16] has the same disadvantage of the AIS algorithm. Another disadvantage is that for each candidate itemset, there are as many entries as its support value.

3. Apriori algorithm

The Apriori algorithm is one of the most popular algorithm in the mining of association rules in a centralized database[1] ,[5]. Agarwal and Srikant in [1] proposed the Apriori Algorithm for finding the frequent itemsets .The Apriori Algorithm proposed to finds frequent items in a given data set using the antmonotone constraint. The name of Apriori is based on the fact that the algorithm uses a prior knowledge of frequent itemset properties. Apriori employs an iterative approach known as a level wise search, where k item sets are used to explore (k+1) itemsets . This algorithm contains a number of passes over the database. During pass k, the algorithm finds the set of frequent itemsets Lk of length k that satisfy the minimum support requirement. Apriori is designed to operate on databases containing transactions. The purpose of the Apriori Algorithm is to find associations between different sets of data. It is sometimes referred to as "Market Basket Analysis". Each set of data has a number of items and is called a transaction. The output of Apriori is sets of rules that tell us how often items are contained in sets of data.

The main drawbacks of Apriori algorithm are

a) It takes more time, space and memory for candidate generation process.

b) To generate the candidate set it requires multiple scan over the database.

4. AprioriTID Algorithm

The AprioriTid algorithm[1] also uses the apriori-gen function to determine the candidate itemsets before the pass begins. The interesting feature of this algorithm is that the database D is not used for counting support after the first pass. It is not necessary to use the same algorithm in all the passes over the data. Apriori still examines every transaction in the database. On the other hand, rather than scanning the database, AprioriTid scans Ck for obtaining support counts, and the size of Ck has become smaller than the size of the database. Based on these

observations AprioriHybrid algorithm has been designed. This uses Apriori in the initial passes and switches to AprioriTid.

5. Apriori Hybrid Algorithm

Apriori performs better than AprioriTid in the initial passes but in the later passes AprioriTid has better performance than Apriori. Due to this reason we can use another algorithm called Apriori Hybrid algorithm [1].In which Apriori is used in the initial passes but we switch to AprioriTid in the later passes. The switch takes time, but it is still better in most cases. It is not necessary to use the same algorithm in all the passes over the data. Apriori still examines every transaction in the database. On the other hand, rather than scanning the database, AprioriTid scans Ck for obtaining support counts, and the size of Ck has become smaller than the size of the database .

6. FP-GROWTH Algorithm

To break the two drawbacks [14] of Apriori algorithm, FP-growth algorithm is used. FP-growth requires constructing FP-tree. For that, it requires two passes. FPgrowth uses divide and conquer strategy. It requires two scans on the database. It first computes a list of frequent items sorted by frequency in descending order (F-List) and during its first database scan. In the second scan, the database is compressed into a FP-tree [15]. This algorithm performs mining on FP-tree recursively. There is a problem of finding frequent itemsets which is converted to searching and constructing trees recursively. The frequent itemsets are generated with only two passes over the database and without any candidate generation process. There are two sub processes of frequent patterns generation process which includes: construction of the FP-tree, and generation of the frequent patterns from the FP-tree. We provide the summary of related work and our findings in table 1.

Table1. The summary of related work and our findings

SN	Author Name	Basic Concept	Claims by Author	Remark
1	Rakesh Agrawal and Ramakrishnan Srikant Year 1994	Association rule	Apriori Algorithm is developed for finding the frequent itemsets	For discovering all significant association rules between items in a large database of sales transactions.
2	Rakesh Agrawal and Ramakrishnan Srikant Year 2000	Reconstruction Algorithm	Reconstruct original shape of data.	Reconstruction procedure to accurately estimate the distribution of original data values.
3	David W. Cheung, Vincent T. Ng, Ada W. Fu, & Yongjian Fu	DMA -Distributed mining of Association rule- algorithm	The performance of DMA depends on the distribution of the data across the partitions	Techniques are used: 1)Candidate Set generation , 2) Local Pruning , 3)Message

	December 1996			Optimization 4)Optimizing partition scanning
4	Murat Kantarcıoglu and Chris Clifton, September 2004	Horizontal and vertical portioning	Private Association Rule Mining And Secure Association Rule Mining	This paper addresses secure mining of association rules over horizontally partitioned data.
5	Jong Soo Park , Ming Syan Chen and Philp S. Yu. Year 1995	algorithm DHP (standing for Direct Hashing and pruning)	DHP proposed has 2 major features: 1 is efficient generation for large itemsets and the other is effective reduction on transaction database size	The issue of mining association rules among items in a large database of sales transactions
6	Rakesh Agrawal and Ramakrishnan Srikant Year 1995	Association Rule	Mining generalized association rules with a large database of customer transactions	To replace each transaction with an "extended transaction" that contains all the items in the original transaction

IV. An approach for mining the association rule in distributed database

We present hereby an approach to find frequent itemsets from the database which is at Different Distributions, and mine the association rules .

- Every site computes the local support counts of all these candidate sets and broad- casts them to the other sites.
- All the sites can then find the globally large itemsets for that iteration.
- Implementation of secure communication between different sites.
- Mining the association rules from the globally large itemsets found in above step

Conclusion

Association rules are basic data mining tools for initial data exploration usually applied to large data sets, seeking to identify the most common groups of items occurring together. There are various association rule mining algorithms. In this paper we have studied and presented several association rule mining algorithms : AIS, SETM, Apriori, Aprioritid , Apriorihybrid, FP-growth. Each algorithm has some advantages and disadvantages. We have also studied and presented data structure used for frequent pattern mining. We have also presented an approach for mining association rule in distributed database.

References

[1] Rakesh Agrawal and Ramakrishnan Srikant, "Fast Algorithms for Mining Association Rules in Large Databases," *Proc 20th International Conference Very Large Data Bases (VLDB)*, pp. 487-499, Year 1994.

[2] Rakesh Agrawal and Ramakrishnan Srikant, "Privacy-Preserving Data Mining," *Proc. ACM SIGMOD Conference*, pp. 439-450, Year 2000.

[3] David.W Cheung, Vincent T. Ng, Ada W. Fu, and Yongjian Fu, "Efficient Mining of Association Rules in Distributed Databases," *IEEE Transaction Knowledge and Data Eng.*, volume no. 8, no. 6, Year December 1996.

[4] Murat Kantarcioglu and Chris Clifton, "Privacy-Preserving Distributed Mining of Association Rules on Horizontally Partitioned Data," *IEEE Transaction Knowledge and Data Eng*., volume no. 16, no. 9, pp. 1026-1037, Year September 2004.

[5] Jong Soo Park, Ming Syan Chen, and Philip S. Yu, "An Effective Hash Based Algorithm for Mining Association Rules," *Proc. ACM SIGMOD Conference*, pp. 175-186, Year 1995.

[6] Rakesh Agrawal and Ramakrishnan Srikant , "Mining Generalized Association Rules," Proc. Int'l Conf. Very Large Data Bases (VLDB), pp. 407-419, 1995

[7] David W. Cheung, Jiawei Han, Vincent T. Ng, Ada W. Fu, and Yongjian Fu, "A Fast Distributed Algorithm for Mining Association Rules," *Proc. Fourth International Conference Parallel and Distributed Information Systems (PDIS*), pp. 31-42, Year 1996

[8] Murat Kantarcioglu, Robert Nix, and Jaideep Vaidya, "An Efficient Approximate Protocol for Privacy-Preserving Association Rule Mining," *Proc. 13th Pacific-Asia Conf. Advances in Knowledge Discovery and Data Mining (PAKDD*), pp. 515-524, Year 2009.

[9] Mihir Bellare, Ran Canetti, and Hugo Krawczyk, "Keying Hash Functions for Message Authentication," *Proc. 16th Ann. International Cryptology Conference Advances in Cryptology (Crypto)*, pp. 1-15, Year1996.

[10] Assaf Schuster, Ran Wolff, and Bobi Gilburd, "Privacy-Preserving Association Rule Mining in Large-Scale Distributed Systems," *Proc. IEEE International Symp. Cluster Computing and the Grid (CCGRID)*, pp. 411-418, Year 2004.

[11] Alexandre Evfimievski, Ramakrishnan Srikant, Rakesh Agrawal, and Johannes Gehrke, "Privacy Preserving Mining of Association Rules*," Proc. Eighth ACM SIGKDD International Conference Knowledge Discovery and Data Mining (KDD*), pp. 217-228, Year 2002.

[12] Tamir Tassa and Ehud Gudes, "Secure Distributed Computation of Anonymized Views of Shared Databases," *Transaction. Database Systems*, volume no. 37, article 11, Year 2012.

[13] Secure Mining of Association Rules in Horizontally Distributed Databases ,Tamir Tassa , IEEE Transactions On Knowledge And Data Engineering, VOL. 26, NO. 4, April 2014.

[14] Sotiris Kotsiantis, Dimitris Kanellopoulos, *AssociationRules Mining: A Recent Overview*, GESTS International Transactions on Computer Science and Engineering, Vol.32 (1), 2006, pp. 71-82

[15] Gagandeep Kaur, Shruti Aggarwal , *Performance Analysis of Association Rule Mining Algorithms*, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 8, August 2013, ISSN: 2277 128.

[16] M. Houtsma and A. Swami. Set-oriented mining of association rules. Research Report RJ 9567, IBM Almaden Research Center, San Jose, Cali- fornia, October 1993.

[17] R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large databases. In Proc. of the ACM SIGMOD Conference on Management of Data, Washington, D.C., May 1993.

[18] H. Grosskreutz, B. Lemmen, and S. R€uping, " Secure Distributed Subgroup Discovery in Horizontally Partitioned Data," Trans. Data Privacy, vol. 4, no. 3, pp. 147-165, 2011.