AN OVERVIEW ON ASSOCIATION RULE MINING ALGORITHMS

Anand V. Saurkar[*]

Abstract— Association Rule Mining is one of the most imperative research areas in the concept of data mining that facilitate the mining of concealed recurrent patterns that based on their own frequencies in the shape of association rules from any item set or datasets containing entities to represent the most recent trends in the given dataset. They have proven to be quite useful in the marketing and retail communities as well as other more diverse fields. In this paper, we present the basic concepts about association rule mining and a brief overview on existing association rule mining algorithms.

Keywords- data mining, association rules.

*****

[*] Department of Computer Science & Engineering, DMIETR, Sawangu (Meghe), Wardha (MH) India

I. Introduction

Data mining is the technique to dig out the inherent information and knowledge from the collection of, incomplete, imperfect ,noisy, fuzzy, random and unsystematic data which is potentially functional and people do not know in advance about this hidden information. The main difference between the traditional data analysis technique such as query reporting and the data mining and is that the data mining is very helpful to determine knowledge and also useful in mining information based on the premise of no clear hypothesis or assumptions. The most important use of data mining is in programmed data analysis technique to come across or to find out earlier unseen or undiscovered associations among various data items in the dataset.

Data mining is the complete analysis step of the Knowledge Discovery in Databases process. It is nothing but a computational activity consisting of discovering meaningful and hidden patterns and information in large datasets of items. Artificial intelligence, machine learning, statistics, and database systems are some of the data mining application areas. In wide-ranging we can say that, the overall aim of the data mining procedure is to dig out meaningful and hidden information from a dataset containing items and then renovate it into a reasonable structure for future use. On other hand, apart from data analysis step, it also involves the concepts of database and data management, data pre-processing. A variety of other activities like inference and complexity considerations, interestingness metrics, and post-processing of discovered structures are also the part of data mining process.

Association Rule Mining (ARM) is one of the most imperative research areas in the concept of data mining that facilitate the mining of concealed recurrent patterns that based on their own frequencies in the shape of association rules from any item set or datasets containing entities to represent the most recent trends in the given dataset.

Association rule mining is to search out association rules that satisfy the predefined minimum support and confidence from given information. The matter is sometimes rotten into two sub problems. Working of first subprogram is used to search out those itemsets whose occurrences exceed a predefined threshold within the database; those itemsets area unit referred to as frequent or massive itemsets and working of second is to get association rules from those massive itemsets with the constraints of lowest confidence. Assume that one of the massive itemsets is Lk, Lk = {I1, I2, … , Ik}, association rules with this itemsets are generated in the following way: the primary rule is {I1, I2, … , Ik-1}→ {Ik}, by checking the confidence this rule are often determined as interesting or not. After analyzing the interest of existing rule, another rule are generated by deleting the last items with in the antecedent and adding it to the consequent, after that the confidences of the new rules are checked to determine the interestingness of them.

These processes are repeated till the antecedent becomes empty [1]. After completion of above process, the first sub-problem is often divided into two sub-problems: candidate large itemsets generation process and frequent itemsets generation process. If supportof the existing itemset exceeds the threshold support then such itemsets are called as large or frequent itemsets and itemsets that are expected or have the hope to be large or frequent are called candidate itemsets.

In several cases, the algorithms generate large number of association rules, usually in thousands or millions. Most of the time association rules become a very large to analyze. End users are unable to understand or validate such large number of complicated association rules, thereby limiting the quality of the data mining results. May methods are plan to reduce the number of association rules, such as generating only "interesting" rules and/or generating only "nonredundant" rules, or generating only those rules satisfying certain other criteria like leverage, coverage, lift or strength.

II. Basic Concepts & Association Rules Algorithms

Association rule mining algorithms can be divided in two basic classes; these are BFS like algorithms and DFS like algorithms [1]. In case of BFS, at first the minimum support is determined for all item sets in a specific level depth, but in DFS, it descends the structure recursively through several depth levels. Both of these can be divided further in two sub classes; these are counting and intersecting. Apriori algorithm comes under the counting subclass of BFS class algorithms. It was the first attempt to mine association rules from a large dataset . The algorithm can be used for both, finding frequent patterns and also deriving association rules from them. FP-Growth algorithm falls under the counting subclass of DFS class algorithms. These two algorithms are the popular example of the classical association rule mining.

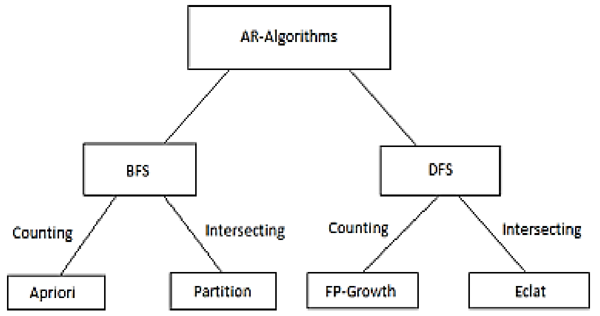

Figure 1: Classification of Mining Algorithm.

Let I=I1, I2, … , Im be a set of m distinct attributes, T be transaction that contains a set of items such that T ⊆ I, D be a database with different transaction records Ts. An association rule is an implication in the form of X→Y, where X, Y ⊂ I are sets of items called itemsets, and X ∩ Y = ϕ. X is called antecedent while Y is called consequent. The rule X→Y means X implies Y.

There are two key measures for association rules, support(s) and confidence (c). Since the database is huge and users concern about only those frequently purchased items, generally thresholds of support and confidence are predefined by users to drop those rules that are not so interesting or functional. Thresholds defined by user for association rules are respectively called as minimal support and minimal confidence. Support(s) of an association rule is defined as the percentage/fraction of records that contain X ∪ Y to the total number of records in the database. E.g., suppose the support of an item is 0.1%, it indicates that percent of the transaction contain purchasing of this particular item is only 0.1 percent.

In association rule mining Confidence is defined as the percentage/fraction of the number of transactions that contain X ∪ Y to the total number of records that contain X.

Confidence is a measure of strength of the association rules, suppose the confidence of the association rule X→Y is 80%, it means that 80% of the transactions that contain X also contain Y together.

In general, a set of items (such as the antecedent or the consequent of a rule) is called an itemset. Length of the itemset is calculated as the number of items in an itemset. Generally, an association rules mining algorithm contains the following steps:

- The set of candidate k-itemsets is generated by 1-extensions of the large (k -1)- itemsets generated in the previous iteration.

- Supports for the candidate k-itemsets are generated by a pass over the database.
- Itemsets that do not have the minimum support are discarded and the remaining itemsets are called large k-itemsets.

Above procedure is repeated until no more large itemsets are found.

The first algorithm projected for mining association rule was The AIS algorithm [2]. In this algorithm only one item consequent association rules are generated. It means that the consequent of those rules only contain one item, e.g. we only generate rules like X ∩ Y→Z but not those rules as X→Y∩ Z. The main drawback of the AIS algorithm is too many candidate itemsets that finally turned out to be small. So that it requires more space and wastes much effort that turned out to be useless. This algorithm also requires too many passes over the whole database.

Apriori is more efficient during the candidate generation process. While guaranteeing completeness, Apriori uses pruning techniques to avoid measuring certain itemsets,.

These are the itemsets that the algorithm can prove will not turn out to be large. On the other hand there are two bottlenecks of the Apriori algorithm. First one is the complex candidate generation process that uses most of the space, memory and time. Another bottleneck is the multiple scan of the database. Afterwards many new algorithms were designed with some modifications or improvements on the basis of Apriori algorithm [1].

III. Efficiency of Association Rule Mining Algorithms

The computational cost of association rules mining can be reduced by following four ways:

- reducing the number of passes over the database
- sampling the database
- adding extra constraints on the structure of patterns
- through parallelization.

In recent years much progress has been made in all these directions.

A. *Reducing the number of passes over the database*

Frequent pattern tree (FP-Tree) mining is another milestone in the development of association rule mining. The most important bottlenecks of the Apriori are break by FP-Tree. Repeated itemsets are generated with only two passes over the database and without any candidate generation process. Frequent length-1 items will have nodes in the tree. Tree nodes are set in such a way that more frequently occurring nodes will have better chances of sharing nodes than less frequently occurring ones. FP-tree is an extended prefix-tree structure used to store critical, quantitative information about frequent patterns [3].

As the support threshold goes down, the length of frequent itemsets will increases radically, FP-Tree work better than Apriori algorithm. The candidate sets that Apriori must handle become extremely large, and therefore the pattern matching with a lot of candidates by searching through the transactions becomes very costly. Two new processes are used by the frequent patterns generation process. First one is the constructing the FP-Tree and second one is the generating frequent patterns from the FP-Tree. The mining result is the same with Apriori series algorithms. Efficiency of FP-Tree algorithm account for three reasons. First the FP-Tree uses only those item which are frequently used, so it is a compressed representation of the original database because and other irrelevant information are removed.

Secondly this algorithm only scans the database twice. At last, as it uses divide and conquer method that significantly reduced the size of the subsequent conditional FP-Tree. Every algorithm has its own restrictions, for FP-Tree it is difficult to be used in an interactive mining system. At the time of interactive mining process, there is provision for users to change the threshold of support according to the rules. However for FP-Tree the changing of support may lead to repetition of the whole mining process. One more drawback is that FP-Tree is that it is not suitable for incremental mining. Since as time goes on databases keep changing and fresh datasets may be inserted into the database. If we uses FP- Tree algorithms, insertions may also lead to a repetition of the whole process.

As of late proposed, TreeProjection is another effective algorithm [4]. The regular thought of TreeProjection is that it builds a lexicographical tree and ventures a substantial database into an arrangement of lessened, thing based sub-databases in view of the frequent pattern mined. The quantity of nodes in its lexicographic tree is precisely that of the frequent itemsets. The productivity of TreeProjection can be clarified by two principle consideration: (1) the transaction projection confines the support counting in a moderately little space; and (2) the lexicographical tree generate the administration and counting of candidates and gives the adaptability of picking proficient technique amid the tree era and transaction projection phrases.

Another algorithm is PRICES, a proficient calculation for mining association rules. Their methodology decreases substantial itemset generation time, known to be the most tedious venture, by checking the database just once and utilizing logical operations as a part of the procedure [5]. Another algorithm for proficient creating huge frequent candidate sets is Matrix Algorithm. The algorithm creates a matrix which entrances 1 or 0 by disregarding the casual database just once, and after that the frequent candidate sets are acquired from the subsequent network [6]. At long last association rules are mined from the frequent candidate sets.

B. *Sampling*

A researcher, Toivonen introduced an association rule mining algorithm which utilize sampling techniques . The methodology can be separated into two stages. In stage 1 a sample of the database is acquired and all relationships in the sample are found. These outcomes are then accepted against the whole database. To boost the viability of the general approach, the creator makes utilization of brought down minimum support on the sample. As the methodology is probabilistic (i.e. subject to the sample containing all the applicable association) not all the tenets may be found in this first pass. Those association that were considered not visit in the sample but rather were frequently visit in the whole dataset are utilized to develop the complete arrangement of association in stage 2 [7].

Another dynamic sampling algorithm, called Sampling Error Estimation (SEE), which expects to distinguish a proper sample size for mining association rules. SEE has two advantages. Initially, SEE is exceedingly proficient in light of the fact that a suitable sampling size can be dead set without the need of executing association rules. Second, the distinguished sample size of SEE is extremely precise, implying that association rules can be profoundly proficiently executed on a sample of this size to get a sufficiently exact result [8].

Particularly, if data comes as a stream streaming at a quicker rate than can be prepared, sampling is by all accounts the main decision. Step by step instructions to sample the data and how huge the sample size ought to be for a given error bound and confidence levels are key issues for specific data mining undertakings. A researcher Li and Gopalan determine the sufficient sample size in light of focal point of confinement hypothesis for sampling huge datasets with substitution [9].

C. *Parallelization*

Association rule revelation procedures have bit by bit been adjusted to parallel frameworks so as to exploit the higher pace and more noteworthy stockpiling limit that they offer[10]. Thetransaction to a conveyed memory framework obliges the apportioning of the database among the processors, a technique that is for the most part completed unpredictably. To build a productivity of association rule mining, other parallelization base algorithm is introduced , which is called as FDM. FDM is a parallelization of Apriori to (imparted nothing machines, each with its own particular part of the database. At each level and on every machine, the database sweep is performed autonomously on the nearby partition. At that point an appropriated pruning procedure is utilized.

Another effective parallel algorithm FPM (Fast Parallel Mining) for mining association rule on an imparted nothing parallel framework has been proposed[11]. It embraces the check appropriation approach and has fused two capable candidate pruning procedures, i.e., disseminated pruning and global pruning. It has a straightforward correspondence plan which performs stand out round of message trade in every cycle. Another algorithm, Data Allocation Algorithm (DAA), is utilizations Principal Component Analysis to enhance the data appropriation before FPM [12].

D. *Constraints based association rule mining*

Numerous data mining strategies comprise in finding pattern frequently happening in the source dataset. Commonly, the objective is to find all the patterns whose frequency in the dataset surpasses a client indicated threshold. Then again, all the time clients need to limit the arrangement of pattern to be found by including additional limitations the structure of pattern. Data mining frameworks ought to have the capacity to endeavor such constraints to speed up the mining procedure. Methods appropriate to constrain driven pattern disclosure can be arranged into the accompanying gatherings:

- post-processing (sifting out pattern that don't fulfill client indicated pattern constraints after the genuine revelation process);
- pattern filtering (coordination of pattern constraints into the real mining process with a specific end goal to create just pattern fulfilling the requirements);

- dataset sifting (confining the source dataset to questions that can potentially contain pattern that fulfill pattern constraints).

A percentage of the researcher concentrate on enhancing the proficiency of constrain based frequent pattern mining by utilizing dataset separating methods. Dataset filtering adroitly changes a given data mining errand into an equal one working on a littler dataset [13]. Rapid Association Rule Mining (RARM) is an association rule mining technique that uses the tree structure to speak to the first database and evades candidate generation process. With a specific end goal to enhance the proficiency of existing mining algorithm, constraints were connected amid the mining methodology to create just those association rule that are intriguing to clients rather than all the association rule [14].

IV. Conclusion

Association rule mining has a wide range of applicability such market basket analysis, medical diagnosis or research, Website navigation analysis, homeland security and so on. In this article, we discuss some of existing association rule mining techniques and algorithms.

The traditional algorithm of association rules discovery proceeds in twosteps. All frequent itemsets are found in the step one . The association rules with the confidence at least *minconf* are generated in the second step. Several different strategies have been discuss to enhance association rule mining efficiency. These techniques are work like reducing the number of passes over the database, adding extra constraints on the structure of patterns, sampling the database, and parallelization.

References

[1] Sotiris Kotsiantis, Dimitris Kanellopoulos, 2006. Association Rules Mining: A Recent Overview. GESTS International Transactions on Computer Science and Engineering, pp. 71-82.

[2] Agrawal, R., Imielinski, T., and Swami, A. N. 1993. Mining association rules between sets of items in large databases. In Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, 207-216.

[3] Han, J. and Pei, J. 2000. Mining frequent patterns by pattern-growth: methodology and implications. ACM SIGKDD Explorations Newsletter 2, 2, 14-20.

[4] Agarwal, R. Aggarwal, C. and Prasad V., A tree projection algorithm for generation of frequent itemsets. In J. Parallel and Distributed Computing, 2000.

[5] Wang, C., Tjortjis, C., PRICES: An Efficient Algorithm for Mining Association Rules, Lecture Notes in Computer Science, Volume 3177, Jan 2004, Pages 352 - 358

[6] Yuan, Y., Huang, T., A Matrix Algorithm for Mining Association Rules, Lecture Notes in Computer Science, Volume 3644, Sep 2005, Pages 370 - 379

[7] Toivonen, H. (1996), Sampling large databases for association rules, in `The VLDB Journal', pp. 134-145.

[8] Chuang, K., Chen, M., Yang, W., Progressive Sampling for Association Rules Based on Sampling Error Estimation, Lecture Notes in Computer Science, Volume 3518, Jun 2005, Pages 505 - 515

[9] Li, Y., Gopalan, R., Effective Sampling for Mining Association Rules, Lecture Notes in Computer Science, Volume 3339, Jan 2004, Pages 391 - 401

[10] Zaki, M. J., Parallel and distributed association mining: A survey. IEEE Concurrency, Special Issue on Parallel Mechanisms for Data Mining, 7(4):14--25, December 1999.

[11] Cheung, D., Xiao, Y., Effect of data skewness in parallel mining of association rules, Lecture Notes in Computer Science, Volume 1394, Aug 1998, Pages 48 – 60.

[12] Manning, A., Keane, J., Data Allocation Algorithm for Parallel Association Rule Discovery,
Lecture Notes in Computer Science, Volume 2035, Page 413-420.

[13] Wojciechowski, M., Zakrzewicz, M., Dataset Filtering Techniques in Constraint-Based Frequent Pattern Mining, Lecture Notes in Computer Science, Volume 2447, 2002, pp.
77-83

[14] Das, A., Ng, W.-K., and Woon, Y.-K. 2001. Rapid association rule mining. In Proceedings of the tenth international conference on Information and knowledge management. ACM Press, 474-481.

[15] Cristofor, L., Simovici, D., Generating an informative cover for association rules. In Proc. of the IEEE International Conference on Data Mining, 2002.

[16] Tien Dung Do, Siu Cheung Hui, Alvis Fong, Mining Frequent Itemsets with Category-
Based Constraints, Lecture Notes in Computer Science, Volume 2843, 2003, pp. 76 - 86