Volume 3, Issue 7, July 2013 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

A Survey on Association Rule Mining Using Apriori Based Algorithm and Hash Based Methods

Priyanka Asthana
Computer Science Deptt.
VIth Sem, BUIT, Bhopal, India

Anju Singh
Information Technology,Deptt.
BUIT, Bhopal, India

Diwakar Singh
Computer Science Deptt.
BUIT, Bhopal, India

Abstract: Association rule mining is the most important technique in the field of data mining. The main task of association rule mining is to mine association rules by using minimum support thresholds decided by the user, to find the frequent patterns. Above all, most important is research on increment association rules mining. The Apriori algorithm is a classical algorithm in mining association rules. This classical algorithm is inefficient due to so many scans of database. And if the database is large, it takes too much time to scan the database. This paper presents many improved Apriori algorithm to increase the efficiency of generating association rules.

Keywords- Data mining, Apriori algorithm, minimum support threshold, multiple scan.

I. Introduction

1.1. Association Rules

Association rule mining is the efficient method which is used in finding the association rules. These association rules describe the associations between the attribute values of any item set. They can be found by means of various methods among which support and confidence [10] , [11] will be considered as the optimized methods in finding them. The key to find the association rules is to find all the frequent item sets present in the given transactional record by means of the minimum support threshold. An association rule is best expressed by means of the expression X -> Y. it means that for any occurrence of item X present in the database there is relatively high probability of occurring the item Y. here X is called as antecedent and Y is called as the consequent. The strength of such rule can only be calculated by means of its support and confidence.

Support

Support 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. Support(s) is calculated by the following formula:

$$\text{Support}(XY) = \frac{\text{Support count of } XY}{\text{Total number of transaction in D}}$$

Support is used to find the strongest association rules in the item sets

Confidence

Confidence is another approach for finding the association rules. Confidence of an association rule is defined as the percentage/fraction of the number of transactions that contain X Y to the total number of records that contain X, where if the percentage exceeds the threshold of confidence an interesting association rule X=>Y can be generated.

$$\text{Confidence}(X|Y) = \frac{\text{Support}(XY)}{\text{Support}(X)}$$

(A) Positive Association Rules

The normal convention in discovering the association rules is by means of any frequent item sets that are present in the given transactional database. The rules that are normally obtained by means of using minimum support threshold and minimum confidence threshold are generally referred as the positive association rules and the rule is of the form ¬A ¬B. That means that they are capable of associating one element to the other element in a given set of transactional records.

(B) Negative Association Rules

Contrary to the positive association rules described above, negative association rules are defmed as the rule that involves the absence of item sets. For example, consider A => ¬B, here , "¬", indicates the absence of an item set B in a set of given transactional records. The rules of the forms (A→ ¬B, ¬A→ B and ¬A→¬B) are negative association rules [12] .

(C) Constraints based association rule mining

In an interactive mining environment, it becomes a necessity to enable the user to express his interests through constraints on the discovered rules, and to change these interests interactively. The most famous constraints are item constraints, which are those that impose restrictions on the presence or absence of items in a rule. These constraints can be in the form of conjunction or a disjunction. Such constraints have been introduced first in [43] where a new method, for incorporating the constraints into the candidate generation phase of the Apriori algorithm, was proposed. In this way, candidates are assured to obey the Item constraints besides the original support and confidence constraints.

Item constraints are not only possible type of rule constraints. Ng et al. [35] presented a wide range of constraints on rules that extends from relational operators on values of the items to constraints on the value of some aggregate functions calculated on the rule items. They defined what is called Constrained Frequent Queries (CAQs) (later named Constrained Frequent-set Queries in [29]) and presented an excellent classification of constraints constructs that can be exploited in them by introducing the notions of succinct and anti-monotone constraints. The CAP (Constrained APriori) waspresented for efficient discovery of constrained association rules.

1.2. Hash Mapping Table (HMT)

Basic Terms

HMT - The data sets of characters in the file mapped to the integer, to enhance the matching effectiveness, and decrease memory footprint.
Hash_ tree - Utilize hash function to build a special data structure for candidate item sets.
Hash_ node - The node of hash tress, it has branch_ node and leaf_ node
Branch_ node - Be used for connect leaf node
Leaf_ node - Be used for connect bucket
Bucket - It has item sets inside HMT(HASH MAPPING TABLE) is a One-dimensional mapping table of the key-value pairs, which bases on hash function, it is intended at compress the data sets, to decrease memory footprint. When the HMT is recognized, it can query the mapped value according to the terms, or in turn. Its main value is a string type of item, and its value is integer. Association rules mining for item sets is all handling its main value.

II. Issues in Finding Association Rules

A. Minimum Support Threshold

Many algorithms such as apriori, FPtree etc. use this minimum support threshold in finding the frequent item sets. This threshold value is pre-set by the users. This value is set by user only. When user set high threshold value any infrequent item sets will lost. And if it is set low, many infrequent item sets will come into consideration. Due to this problem an optimized decision cannot be taken. So threshold should be set very precisely.

B. Multiple Scans across the Transactional Database

While finding any frequent item sets, we have to scan whole database many times. This multiple scan will lead to following problems:

i) Wastage of time, because searching entire database for any item takes lot of time.
ii) Wastage of space, because lot of memory is needed.

C. Performance on Scaling

If the no of transactions increased, performance is not scaled with increasing transactions. Scalability is an important factor which is difficult to implement with algorithms of association rule mining.

In this paper we are trying to remove these problems, as well as include database security.

III. Literature Survey

3.1. Apriori Algorithm

Apriori algorithm was first proposed by Agrawal in [20].Apriori is more efficient during the candidate generation process [19]. It uses a breadth-first search strategy [25] to count the support of item sets and uses a candidate generation function which exploits the downward closure property of support. Apriori uses pruning techniques to avoid measuring certain item sets, while guaranteeing completeness. The Apriori algorithm is based on the Apriori principle [24], which says that the item set X' containing item set X is never large if item set X is not large. Based on this principle, the Apriori algorithm generates a set of candidate large item sets whose lengths are (k+1) from the large k item sets (for $k \geq 1$) and eliminates those candidates, which contain not large subset. Then, for the rest candidates, only those with support over minimum support threshold are taken to be large (k+1)-item sets. The Apriori generate item sets by using only the large item sets found in the previous pass, without considering the transactions.

The Apriori algorithm takes advantage of the fact that any subset of a frequent item set is also a frequent item set. The algorithm can therefore, reduce the number of candidates being considered by only exploring the item sets whose support count is greater than the minimum support count. All infrequent item sets can be pruned if it has an infrequent subset. In the process of finding frequent item sets, Apriori avoids the effort wastage of counting the candidate item sets that are known to be infrequent. The candidates are generated by joining among the frequent item sets level-wisely, also candidate are pruned according the Apriori property. As a result the number of remaining candidate item sets ready for further support checking becomes much smaller, which dramatically reduces the computation, I/O cost and memory requirement. The Apriori algorithm uses a bottom-up breadth-first approach to find the large item set. In Apriori,in each iteration (or each pass) it creates a candidate set of large item sets, counts the number of occurrences of each candidate item set, and then decides large item sets based on a pre-determined minimum support. In the first iteration, Apriori simply scans all the transactions to count the number of occurrences for each item. Minimum support and minimum confidence are two important indicators of association rules. The algorithm uses apriori principle to generate candidate k-item sets from frequent (k-1)-item sets, and prunes candidate item sets. Through the support counting, get candidate k-item sets. Then the candidate k-item sets generate frequent (k-1)-item sets, so back and forth, until the frequent item sets cannot be produced.

3.2. MSApriori

Association rule mining is an important model in data mining. Its mining algorithms discover all item associations (or rules) in the data that satisfy the user-specified minimum support (minsup) and minimum confidence (minconf) constraints. Since only one minsup is used for the whole database, the model implicitly assumes that all items in the data are of the same nature and/or have similar frequencies in the data. This is, however, seldom the case in real life applications. In many applications, some items appear very frequently in the data, while others rarely appear. If minsup is set too high, those rules that involve rare items will not be found. To find rules that involve both frequent and rare items, minus has to be set very low. This may cause combinatorial explosion because those frequent items will be associated with one another in all possible ways. This dilemma is called the rare item problem. The MSApriori Technique allows the user to specify multiple minimum supports to reflect the natures of the items and their varied frequencies in the database. In this model, the minimum support of a rule is expressed in terms of minimum item supports (MIS) of the items that appear in the rule. That is, each item in the database can have a minimum item support specified by the user. By providing different MIS values for different items, the user effectively expresses different support requirements for different rules.

3.3. MCISI Algorithm

A sporadic rule is an association rule which has low support but high confidence. In general, sporadic rules are of rare occurrence but high value in many cases. Of the two types of perfectly and imperfectly sporadic rules, imperfectly sporadic rules are more difficult to mine. Until now the problem of mining imperfectly sporadic rules has not been solved completely. To discover imperfectly sporadic rules, in [32-33] the authors have divided the imperfectly sporadic rules into four types including: (1) rules have both frequent and infrequent item sets in its antecedent and consequent; (2) rules have only frequent item sets in both its antecedent and consequent; (3) rule have only frequent item sets in its consequents and infrequent item sets in its antecedents; and (4) rules have only infrequent item sets in its consequents and frequent item sets in its antecedents. The authors then proposed the MIISR algorithm to discover rules of the type 3 above. Mining imperfectly sporadic rules of the types 1 and 2 is open while meaning of imperfectly sporadic rules of the type 4 is very poor. Search space for mining imperfectly sporadic item sets with two thresholds includes item sets, which are created from the set of frequent items for maxSup in combination with themselves and with the set of infrequent items for maxSup but frequent for minSup. Based on the search space, the MCISI algorithm will find closed imperfectly sporadic item sets with two thresholds under the approach of the CHARM algorithm [39]. It performs a search over a novel Itemset-Tidset search space by removing all item sets which are not imperfectly sporadic item sets with two thresholds and are not closed. The MCISI algorithm was proposed to find closed imperfectly sporadic item sets with two thresholds. This will enable the finding of many imperfectly sporadic rules which cannot be done by other algorithms such as MIISR algorithm.

3.4. Aprori with Systematic rules

Systematic algorithm has been proposed in this paper[1]. In this algorithm, the user is not allowed to specify any minimum support threshold values to find the frequent patterns; instead the system itself generates he minimum threshold values, thus plugging the loophole of other algorithms. Using this approach, the user is well aware of entire information aiding him to take correct informed decisions. They also introduce the concept of timing algorithm along with the systematic algorithm, which will statically assign a unique value to each record of the transactional database. This technique is mainly used to save time by scanning through the entire transactional database only once rather than making multiple scans. The benefit of one scan database leads to better performance and minimization of time and with the help of systematic algorithm to discover patterns using highest support values in the systematic table. This will take any dataset as input, and a systematic table is constructed for every transaction in the provided in the dataset.

The systematic tables for every item sets involved in the datasets are calculated by the following conditions -

Supp (A---> B) = supp (A) + supp (B) + supp (A UB) ……(1)

Supp (A--->⌐B) =supp (A) - supp (A UB) …..... (2)

Supp (⌐A---> B) =supp (B) - supp (A UB) ……(3)

Supp (⌐A---> ⌐B) =1- supp (A) - supp (B) + supp(A UB) …… (4)

3.5. HMT (Hash Mapping Table)

(a) Building process

1. Take the data set line by line, according to the list separator to separate the items.
2. Inquire the HMT whether it has that item.
3. If the item already present, ignores it; or else, adds it into the HMT through the hash.

The above process can incorporate with the compression of data set, the incorporated process is following:

1. Take the data set line by line, according to the list separator to separate the items.
2. Inquire the HMT whether it has that item.
3. If the item already present, ignores it; or else, adds it into the HMT through the hash.
4. At the same time, include the compressed data into the memory map of the data set.

(b) Processing and analysis HMT

In the Apriori algorithm, when utilize the HMT to compress the data sets, the processing of item sets have become the treatment of integer data. That incorporates the comparison of item sets, HASH computer, the generation of subsets etc. When compress the item sets, it will expend some handle time, but the time of support counting in Apriori is more than the compress time.

IV. Conclusion and Future Scope

Association mining rules are very useful in applications going beyond the standard market basket alanysis. We have shown here various Apriori algorithms used to find frequent items in a given transaction of database. Since Apriori algorithm was first introduced and as experience was pile up, there have been many attempts to devise more efficient algorithms of frequent itemset mining. Many algorithms share the same idea with Apriori in that they generate candidates. These include hash-based technique, partitioning, sampling and using vertical data format. Hash-based technique can minify the size of candidate itemsets. Further hash based methods can be combined with Apriori algorithm to reduce time and space complexity.

References

[1] R. Agrawal. T. Imielinski. and A Swami. "Mining Association Rules between Sets of Items in Large Databases," Proc. 1993 ACM SIGMOD Int'I Conf. Management of Data (SIGMOD '93), pp. 207-216, 1993.

[2] R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules," Proc. 20th Int'I Conf. Very Large Data Bases, pp. 487-499, 1994.

[3] R. Sri kant and R. Agrawal, "Mining Generalized Association Rules," Future Generation Computer Systems, vol. 13, nos. 2/3, pp.161-180, 1997.

[4] Cohen, E.; Datar, M.; Fujiwara, S.; Gionis, A; Indyk, P.; Motwani, R.; Ullman, 1. D. & Yang, C., "Mining association rules with multiple minimum supports", ACM, 1999, pp. 337-341.

[5] Cohen, E.; Datar, M.; Fujiwara, S.; Gionis, A; Indyk, P.;Motwani, R.; Ullman, 1. D. & Yang, c., "Finding Interesting Associations without Support Pruning", fCDE '00: Proceedings of the 16th International Conference on Data Engineering, IEEE Computer Society, 2000, 489.

[6] Huang, Y.; Xiong, H.; Shekhar, S. & Pei, J., "Mining confident co-location rules without a support threshold", SAC '03: Proceedings of the 2003 ACM symposium on Applied computing, ACM, 2003, 497-501.

[7] Koh, Yun Sing and Rountree, Nathan., "Finding Sporadic Rules Using Apriori-Inverse", Lecture Notes in Computer SCience, Springer Berlin 1 Heidelberg, 2005, Vol. 3518" pp. 97-106.

[8] Koh, Y. S.; Rountree, N. & ODKeefe, R., "Mining Interesting Imperfectly Sporadic Rules", Lecture Notes in Computer Science, Springer Berlin 1 Heidelberg, 2006, 3918, 473-482.

[9] Brin, S.; Motwani, R. & Silverstein, C., "Beyond market baskets:generalizing association rules to correlations", SlGMOD Rec., ACM, 1997,26,265-276.

[10] J.S. Park, M. Chen, and P.S. Yu, "An Effective Hash Based Algorithm for Mining Association Rules," Proc. 1995 ACM SIGMOD Int'l Conf. Management of Data, ACM Press, 1995.

[11] J.W. Han, M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann, San Francisco, (200 I).

[12] Srikant, R. & Agrawal, R., "Mining quantitative association rules in large relational tables", SfGMOD Rec., ACM, 1996,25, 1-

[l3] R. Forsyth, "Zoo Data Set," Orange, AI Lab, http://magix.fri.uni-Ij.silorange/doc/datasets/zoo.htm, 1990.

[14] S.-J. Yen and Y.-S. Lee, "Mining Interesting Association Rules:A Data Mining Language," Advances in Knowledge Discovery and DataMining, pp. 172-176, Springer, 2002.

[I5] A Das, D.K. Bhattacharyya, and J.K. Kalita, "Horizontal versus Vertical Partitioning in Association Rule Mining: A Comparison," Proc. Sixth Int'l Conf. Computational Intelligence and Natural Computation (CINC), pp. 1617-1620,2003.

[16] S. Chiu, W.-k. Liao, and A Choudhary, "Design and Evaluation of Distributed Smart Disk Architecture for IO-lntensive Workloads," Proc. Int'I Conf. Computational Science (ICCS '03), pp. 230-241, 2003.

[17] S. Brin, R. Motwani, and C. Silverstein, "Beyond Market Baskets: Generalizing Association Rules to Correlations," Proc. 1997 ACM SIGMOD, pp. 265-276, 1997.

[18] Association Rule Mining: A Survey: http://sci2s.ugr.es/keel/pdf/specific/report/zhao03ars.pdf

[19] Agrawal et al. 1993

[20] Agrawal, R. and Srikant, R. 1995. Mining sequential patterns, P. S. Yu and A. S. P. Chen, Eds. IEEE Computer Society Press, Taipei, Taiwan, 3{14.
[21] Han et al. 2000
[22] Han, J. and Kamber, M. 2000. Data Mining Concepts and Techniques. Morgan Kanufmann.
[23] Agrawal, R., Imielinski, T., and Swami, A. N. 1993. Mining association rules between sets of items in large databases.
[24] http://www.users.cs.umn.edu/~kumar/dmbook/ch6.pdf -Association analysis Basic concepts and algorithms.
[25] Agrawal, Rakesh; and Srikant, Ramakrishnan; Fast algorithms for mining association rules in large databases
[26] http://chemeng.utoronto.ca/~datamining/dmc/association_rules.htm
[27] http://en.wikibooks.org/wiki/Data_Mining Algorithms_In_R/Frequent_Pattern_Mining/The_FPGrowth_Algorithm#FP-Tree_structure
[28] http://software.intel.com/en-s/articles/Multicoreenabling- FP-tree-Algorithm-for-Frequent-Pattern-Mining
[29] Agrawal R., and Srikant R.: Fast Algorithms for Mining Association Rules. Proc. Very Large Database International Conference, Santiago, pp. 487–498, 1994.
[30] Agrawal R., Mannila H., Srikant R., Toivonen H., and Inkeri Verkamo A.: Fast Discovery of Association Rules. Advances in Knowledge Discovery and DataMining. The MIT Press, pp.307-328,1996.
[31] Koh Y. S., and Rountree N.: Finding Sporadic Rules Using Apriori-Inverse. PAKDD 2005, LNAI 3518, pp 97-106, 2005.
[32] Koh Y. S., and Rountree N.: Finding Interesting Imperfectly Sporadic Rules. PAKDD 2006, pp 473-482, 2006.
[33] Koh Y. S., Rountree N., and O'Keefe R. A.: Mining Interesting Imperfectly Sporadic Rules. Knowledge and Information System; 14(2), pp179-196, 2008.
[34] Koh Y. S., and Rountree N.: Rare Association Rule Mining via Transaction Clustering. The Seventh Australasian Data Mining Conference (AusDM 2008).
[35] Kiran R. U., and Reddy P. K.: An Improved Multiple Minimum Support Based Approach to Mine Rare Association Rules. http://www.iiit.net/techreports/2009_24.pdf
[36] Ling Zhou, and Stephen Yau: Association Rule and Quantitative Association Rule Mining among Infrequent Items. MDM'07, August 12, 2007, San Jose, California, USA.
[37] Pasquier N., Bastide Y., Taouil R., and Lakhal L.: Efficient Mining of Association Rules Using Closed Itemset Latics. Information Systems, Vol 24, No. 1, pp. 20-46, 1999.
[38] Szathmary L., Napoli A., and Valtchev P.: Towards Rare Itemset Mining. In Proceedings of the 19th IEEE international Conference on Tools with Artificial Intelligence - Vol.1 (ICTAI 2007) - Volume 01(October 29 - 31, 2007). ICTAI. (pp. 305-312). Washington, DC: IEEE ComputerSociety
[39] Zaki M. J., and Hsiao C.: CHARM: An Efficient Algorithm for Closed Association Rule Mining. In Proceedings, SIAM-02 International Conference on Data Mining, 2002.
[40] Zaki M. J.: Mining Non-Redundant Association Rules. Data Mining and Knowledge Discovery, 9, 223–248, 2004.
[41] UCI-Machine Learning Repository, http://archive.ics.uci.edu/ml/datasets.html

