Distributed Association Rule Mining Algorithms - A Review

Sowmyashree[1] Bhavna Arora[2] Shweta Sharma[3] Tejal Rachh[4]
[1,2,3,4]Assistant Professor
[1,2]Department of Information Technology [3,4]Department of Computer Engineering
[1,2,3,4]Atharva College of Engineering, Mumbai, India

Abstract— Association rule mining is a large data mining research area. Association rule mining is most popular and well researched practice for determining interesting relations between variables in large databases. Most Association rule mining (ARM) algorithms cater to a centralized environment. This paper describes a review of different association rule mining algorithms and their limitations. This paper also proposes a Distributed Count Association Rule Mining Algorithm (DCARM), for distributed systems, which can be experimented on real time data sets. Proposed system can eventually help in sales forecast and making business plans accordingly to suit the market and hence raise the profits accordingly.

Key words: Association Rule Mining, DARM, Market Basket Analysis, Apriori

I. Introduction

Association rule mining is a well-researched and popular method for discovering interesting relations among variables in large databases. This technique is proposed to identify strong rules discovered in databases using different measures of interestingness. The authors Rakesh Agrawal et al.[1] introduced association rules for determining regularities between products in large-scale transaction data documented by point-of-sale (POS) systems in supermarkets.

For example, the rule {onions, potatoes} →{burger} found in the sales data of a supermarket would indicate that if a customer purchases onions and potatoes together, he or she is likely to also purchase hamburger meat. Such information can be used as the basis for decisions about marketing actions such as, e.g., promotional pricing or product placements. As an addition to the above example from market basket analysis association rules are engaged today in many application areas including intrusion detection, Web usage mining, and Bioinformatics and Continuous production. As opposed to sequence mining, association rule learning usually does not consider the order of items either within a transaction or across transactions [2].

The problem of association rule mining can be defined as: Let I={i_1, i_2,i_n} be a set of n binary attributes called items. Let D = {t_1, t_2_m} be a set of transactions called the database. Each transaction in D has a unique transaction ID and comprises a subset of the items in I. Rule is defined as an inference of the form X →Y. The sets of items (for short itemsets) X and Y are called antecedent (left-hand-side) and consequent (right-hand-side or RHS) of the rule respectively [3].

Support(s) and confidence(c) are the two important basic measures for association rules. Since the database is huge and users concern about only those frequently purchased items, typically thresholds of support and confidence are predefined by users to drop those rules which are not so useful. The two thresholds are termed minimal confidence and minimal support respectively.

Support(s) of an association rule is defined as the percentage of records that comprise $X \cup Y$ to the total number of records in the database. Suppose the support of an item is 0.1%, it means only 0.1 percent of the transaction contain buying of this item.

Confidence of an association rule is definedas the percentage of the number of transactions that comprise $X \cup Y$ to the total number of records that contain X. Confidence is a amount of strength of the association rules, suppose the confidence of the association rule $X \Rightarrow Y$ is 80%, it implies that 80% of the transactions that contain X also contain Y together.

In general, a set of items (antecedent or the consequent) is called an *itemset*. The number of items in an itemset is called the length of an itemset. Itemsets of some length k are denoted to as *k-itemsets.*

II. Literature review

Many algorithms were introduced for discovering association rules. AIS was the first algorithm proposed for mining association rules. Various researches were done to improve the performance and scalability of Apriori.

A. AIS Algorithm

The AIS (Agrawal, Imielinski, Swami) algorithm [4] focuses on refining the quality of databasesin addition with necessary functionality to solve decision support queries. In this algorithm only one item consequent association rules are produced, which means that the consequent of those rules only contain one item, for example we only generate rules like $X \cap Y \Rightarrow Z$ but not those rules as $X \Rightarrow Y \cap Z$. The databases were scanned many times to get the frequent item sets in AIS.

Steps involved in AIS algorithm are as follows:

1) Candidate item sets will be generated and will be counted on-the-fly as the database is scanned.
2) For each transaction, it is determined which of the large item sets of the preceding pass are contained in this transaction.

3) New candidate item sets will be generated by extending large item sets with other items in this transaction.

1) Limitations of AIS algorithm are:

1) Too many candidate item sets that lastly turned out to be small will be generated, which requires more space and wastes much effort that turned out to be useless.
2) AIS algorithm results in unnecessarily generate and count too many candidate item sets that turn out to be small.
3) This algorithm requires too many passes over the whole database.

B. Apriori Algorithm

Apriori principle states that, if an item set is frequent, then all of its subsets must be frequent. The Apriori algorithm is built on the Apriori principle, which states that the item set X' containing item set X is never large if item set X is not large. Based on above principle, the Apriori algorithm generates a set of candidate large item sets having lengths (k+1) from the large k item sets (for k ≥1) and eliminates those candidates, which encompass not large subset. Then, for the remaining candidates, only those with support above minimum support threshold are considered to be large (k+1)-item sets. The Apriori generate item sets by using only the large item sets found in the preceding pass, without considering the transactions. Apriori algorithm takes benefit of the fact that any subset of a frequent item set is also a frequent item set. Therefore, the algorithm can reduce the number of candidates being considered by only exploring the item sets whose support count is larger than the minimum support count.

Steps involved in Apriori algorithm are as follows:

1) Candidate item sets are generated using only the large item sets of the previous pass without considering the transactions in the database.
2) The large item set of the previous pass is joined with itself to generate all item sets whose size is higher by 1.
3) Each generated item set that has a subset which is not large is deleted. The remaining item sets are the candidate ones.

1) Limitations of Apriori algorithm are:

- It takes more time, space and memory for candidate generation process.
- To generate the candidate set it requires multiple scan over the database.
- Uses a uniform minimum support threshold.
- Difficulties to find rarely occurring events.

C. AprioriTid Algorithm

The AprioriTid algorithm also makes use of the apriori-gen function to determine the candidate itemsets earlier the pass begins. The remarkable feature of this algorithm is that the database D is not used for counting support after the first pass. There is no necessity to use the same algorithm in all the passes over the data. Apriori still examines every transaction in the database. On the other hand, instead of scanning the database, AprioriTid scans Ck for gaining support counts, and the size of Ck has become lesser than the size of the database

Steps involved in Aprioritid Algorithm are:

1) The database is not used at all for counting the support of candidate item-sets after the first pass.
2) The candidate item-sets are generated in the similar way as in Apriori algorithm.
3) Another set C' is generated of which every member is having the TID of each transaction and the large item-sets present in this transaction. This set is used to count the support of each candidate item-sets.

1) Limitations of AprioriTid algorithm are:

1) An extra cost is incurred when shifting from Apriori to AprioriTid.
2) Suppose at the end of K th pass we decide to switch from Apriori to AprioriTid. Then in the (k+1) pass, after having generated the candidate sets will have to add the Tids to C'k+1.

D. Apriorihybrid Algorithm

It is not essential to use the same algorithm in all the passes over the data. Apriori still examines all transaction in the database. But, on the other hand, rather than scanning the database, AprioriTid scans Ck for obtaining support counts, and the size of Ck has become lesser than the size of the database. Apriori performs well than AprioriTid in the initial passes but in the late passes AprioriTid has better performance than Apriori. Due to this reason another algorithm called Apriori Hybrid algorithm can be used. Based on these observations AprioriHybrid algorithm has been designed [5].

III. PROPOSED WORK

This paper proposes a Distributed Count Association Rule Mining Algorithm that enables clients (Manager, Administrators, Executives and Decision Makers) to analyze the market products. We basically, predict the sale by tracking and analyzing the usage pattern. Using a DCARM enables the use of large amount of customer transaction. Using the concept of association rule mining, useful data that is knowledge can be mined, here, in the project, data is market basket analysis. Since a market data analysis is very booming field, market basket analysis has got upper hand in the domain. An efficient algorithm is used to find the desired information resources and their usage pattern and to reduce communication cost and communication overhead of geographically spread data. It has become increasingly necessary for users to employ automated tools in find the desired information resources, and to track and analyse their usage patterns.

A. DCARM Algorithm

The algorithm involves the following steps:

1) Divide the database evenly into horizontal partitions among all processes.
2) Every process scans its local database partition to gather the local count of each item.
3) Every processes exchange and sum up the local counts to get the global counts of all items and find frequent 1-itemsets.
4) Then set level k = 2.
5) Every processes generate candidate k-itemsets from the mined frequent (k-1)-itemsets.
6) Every process scans its local database partition to collect the local count of each candidate k-itemset. At the same time, DCARM also eliminates all globally infrequent (k-1)-itemsets from every transaction and inserts the new transaction (that is, a transaction without infrequent (k-1)-itemset) into memory.
7) All processes exchange and sum up the local counts into the global counts of every candidate k-itemsets and find frequent k-itemsets among them.
8) Repeat 5 - 8 with k = k + 1 until no more frequent itemsets are found.

B. Benefits over Existing Technologies

The various benefits to the banks are listed as follows:-

1) Distributed Technology

Distributed Count Association Rule Mining Algorithm has overcome the drawback of the apriori algorithm and has provided the functionality of Distribution i.e., this algorithm can be connected between various workstations in a network. Whereas Apriori algorithm basically works on a single workstation and doesn't provide the functionality to connect with the other workstation in a network. Also nothing physical needs to be sold to the customers. It is all secure software that needs to be installed at a store workstation.

2) Enhanced Security

Provides a secure, store-based and user friendly alternative to the non-secure apriori algorithm. Every store has assigned a store_id and password which can only be accessed by respective store manager.

3) Wider Customer Base

The ability to offer existing store additional functional services by improving their access to, and control over, their store transaction information. Ensure a better inventory control systems by keeping the track of the sales report.

The various benefits to the customer are listed as follows:-

- No Additional Hardware cost: Stores only needs to install the application so no additional hardware is required.
- Convenience:Available anywhere and anytime.
- Ease of Use**:** Easy-to-understand, easy-to-use "Market Basket Analysis" concept. Conduct all transactions with a few clicks and keep track of sales and accounts.

The performance of DARM algorithms can be increased by focusing on two major issues communication and synchronization. Communication is one of the most important DARM objectives. DARM algorithms will perform better if we can decrease communication (for example, message exchange size) costs. Synchronization insists each participating site to wait for a certain period until globally frequent itemset generation completes. Each site will delay longer if computing support counts takes extra time. Hence, we reduce the computation time of candidate itemsets' support counts. To reduce communication costs, several message optimization techniques can be used.

IV. APPLICATIONS

The principle application of Distributed Count Association Rule Mining Algorithm is that it can be used in sales prediction system (i.e., which products are bought together and which products are associated with one another). Managers can use the system to predict the sales and accordingly design their business plans

It provides for placement of items or products in retail stores. It can be used to simplify the process of marketing by identifying the products which are associated with one another. It helps in Inventory Control System. It also proves to be helpful for customers during online shopping.

V. CONCLUSION

With the explosive evolution of information sources available on the World Wide Web, it has become increasingly necessary for users to employ automated tools to find the desired information resources, and also to track and analyse their usage patterns. This paper discussed various association rule mining algorithms and their limitations. This paper proposes a distributed association rule mining algorithm by utilizing the global count on frequent itemsets for distributed systems. This algorithm is an efficient method for generating association rules from different datasets, distributed among various sites.

REFERENCES

[1] 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.

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

[3] Sharma, L.K., Vyas, O.P., Tiwary, U.S., Vyas, R. A Novel Approach of Multilevel Posi- tive and Negative Association Rule Mining for Spatial Databases, Lecture Notes in Com- puter Science, Volume 3587, Jul 2005, Pages 620 - 629

[4] 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.

[5] Verma, K., Vyas, O.P., Vyas, R., Temporal Approach to Association Rule Mining Using T-Tree and P-Tree, Lecture Notes in Computer Science, Volume 3587, Jul 2005, Pages 651 - 659