Association Rule Mining for Identifying Optimal Customers Using MAA Algorithm

Vasudha Rani Vaddadi
Department of Information Technology
GMR Institute of Technology, Rajam, Andhra Pradesh, India

Abstract— identifying customers which are more likely potential for a product and service offering is an important issue. In customers identification data mining has been used extensively to predict potential customers for a product and service. Most of the research effort in the scope of association rules has been oriented to simplify the rule set and to improve performance of the algorithm. With the recent advancement of Internet and Web Technology, web search has taken an important role in the ordinary life. This project suggests a new framework of algorithm MAA that overcomes the limitations associated with existing methods and enables the finding of association rules based on Apriori Algorithm among the presence and/or absence of a set of items without a preset minimum support threshold and Minimizing Candidate Generation.

Key words: Apriori, Improved Apriori, Frequent item set, Support, Candidate item set, Time consuming

I. INTRODUCTION

Data mining is the process of analyzing data from different perspectives and summarizing it into useful information - information that can be used to increase revenue, cuts costs, or both. Association rule mining, one of the most important and well researched techniques of data mining. It aims to extract interesting correlations, frequent patterns, associations or casual structures among sets of items in the transaction databases or other data repositories. Association rules are widely used in various areas such as telecommunication networks, market and risk management, inventory control etc. Various association mining techniques and algorithms will be briefly introduced and compared later. Association rule mining is to find out association rules that satisfy the predefined minimum support and confidence from a given database. The problem is usually decomposed into two sub problems. One is to find those item sets whose occurrences exceed a predefined threshold in the database; those itemsets are called frequent or large itemsets. The second problem is to generate association rules from those large item sets with the constraints of minimal confidence.

The basic objective of finding association rules is to find all co-occurrence relationships called associations. Since it was first introduced in1993 by Agrawal et. al, it has attracted a great deal of attention. Many efficient algorithms, extensions and applications have been reported.The classic application of association rule mining is market basket data analysis, which aims to discover how items purchased by associated. Association rules are of form X -> Y,where X and Y are collection of items and intersection of X and Y is null. For example it may find that "95 percent of customers who bought bread (X) also bought milk (Y)" A rule may contain more than one item in the antecedent and consequent of the rule. Every rule must satisfy two users specified constrains: one is the measure of statistical significance called support and the other is a measure of goodness called confidence.

Computers and software play an integral part in the working of businesses and organizations. Animmense amount of data is generated with the use of software. These large datasets need to be analyzed for useful information that would benefit organizations, businesses and individuals by supporting decision making and providing valuable knowledge. Data mining is an approach that aids in fulfilling this requirement. Data mining is the process of applying mathematical, statistical and machine learning techniques on large quantities of data (such as adata warehouse) with the intention of uncovering hidden patterns, often previously unknown. Data mining involves three general approaches to extracting useful information from large data sets, namely, classification, clustering and association rule mining. This paper elaborates upon the use of association rule mining in extracting patterns that occur frequently within a dataset and showcases the implementation of the Apriori algorithm in mining association rules from a dataset containing sales transactions of a retail store.

Existing System are Association rule mining can be an important data analysis method to discover associate rules in CRM. The Apriori algorithm is a proficient algorithm for determining all frequent customers in CRM. But these are not the only problems that can be found and when rules are generated and applied in different domains. Troubleshooting for them should also take into consideration the purpose of association model and the data they come from. Some of drawbacks like non interesting rules, low algorithm performance arts are found in the algorithm. Several past studies addressed the problem of mining association rules with different Supports will not be appropriate in large dataset and they cannot generate more useful rules.

A. Disadvantages

- Large Number of in-frequent item sets are generated which increase the space complexity
- Too many database scans are required because large number of itemsets are generated.
- As the number of database scans are more the time complexity increases as the database increases.

The Proposed System is an efficient algorithm for generating frequent Item sets and is optimized to takes less time compare to the existing algorithms. The main aim of this algorithm is to reduce execution time and memory utilization as compared to the existing algorithms. The framework has been tested on several datasets. By using association rule mining, the profit and frequency value of each customer is computed. Based on the mining result, the companies provide offers to customer using swarm intelligence technique known as particle swarm optimization. This offer does not affect the company revenues as well as satisfying the customers. This process will make a good relationship between the customers

and organizations and to satisfy the customers forever with company's rule.

B. Advantages

- Performance of modified algorithm has been compared with the FP growth, Dn FP growth and Apriori.The run time is the time to mine the frequent item sets.
- Execution time for the MAA algorithm is constant for a certain data set when the support factor decreases from 40% to 5% while, at the same time, the execution time of the MAA increases dramatically. For a support factor of 30% or greater and a data set of 40,000 transactions.

C. Mining Association Rules between Sets of Items in Large Databases

In this paper, several organizations have collected massive amounts of such data. These data sets are usually stored on tertiary storage and are very slowly migrating to database systems. One of the main reasons for the limited success of database systems in this area is that current database systems do not provide necessary functionality for a user interested in taking advantage of this information. This paper introduces the problem of \mining" a large collection of basket data type transactions for association rules between sets of items with some minimum specified confidence, and presents an efficient algorithm.

II. ALGORITHM

The modified Apriori algorithm reduces the number of database scans and the redundancy while generating subtests and verifying them in the database. This algorithm needs to scan the database only once and also does not require to find the candidate set when searching for frequent item set.

D- A information of transaction Min_sup- the minimum support count threshold

1) Step1: Within the initial iteration of the algorithmic rule, every item may be a member of the set of candidate 1-itemset C1. The algorithmic rule merely scans all the transaction to count the quantity of occurrences of every item.
2) Step2: The set of frequent item sets, L1, is set by comparing the candidate count with minimum support count that contains candidate 1-itemsets satisfying minimum support.
3) Step3: To come up with the set of frequent 2-itemsets, L2, the algorithmic rule generates a candidate set of a pair of-item set and so the transactions in D are scanned and therefore the support count of every candidate item set in C2 is accumulated and so repetition of the step2.
4) Step4: Then D2 is set from L2.
5) Step5: Generate C3 candidates from L2 and scan D2 for count of every candidate.
6) Step6: At the tip of the pass, verify that of the candidate item sets are literally massive, and people become the seed for following pass.
7) Step7: This method continues till no new massive item sets are found

Apriori Algorithm is level by Search. A large number of association rule mining algorithms have been developed with different mining efficiencies. Any algorithm should find the same set of rules though their computational efficiencies and memory requirements may be different. The best known mining algorithm is Apriori algorithm. The Apriori algorithm works in two steps:

a. Generate all frequent itemsets: A frequent item set is an itemset that has transaction support above minimum support.
b. Generate all confident association rules from frequent itemsets: A confident association rule is a rule with confidence above minimum confidence.

A. Support

The support of a rule, X → Y, is the percentage of transactions in T that contains X∪ Y, and can be seen as an estimate of the probability, Pr(X∪ Y). The rule support thus determines how frequent the rule is applicable in the transaction set T. Let n be the number of transactions in T. The support of the rule X → Y is computed as follows:

Support = (X ∪ Y). Count /n Support is a useful measure because if it is too low, the rule may just occur due to chance.

Furthermore, in a business environment, a rule covering too few cases (or transactions) may not be useful because it does not make business sense to act on such a rule (not profitable).

B. Confidence

The confidence of a rule, X→ Y, is the percentage of transactions in T that Contain X also contain Y. It can be seen as an estimate of the conditional probability, Pr(Y | X). It is computed as follows:

Confidence = (X ∪ Y).count /X .count Confidence thus determines the predictability of the rule. If the confidence of a rule is too low, one cannot reliably infer or predict Y from X. A rule with low predictability is of limited use.

III. SYSTEM ARCHITECTURE

As the complexity of systems increases, the specification of the system decomposition is critical. Moreover, subsystem decomposition is constantly revised whenever new issues are addressed. Subsystems are merged into alone subsystem, a complex subsystem is split into parts, and some subsystems are added to take care of new functionality. The first iterations over the subsystem decomposition can introduce drastic changes in the system design model.

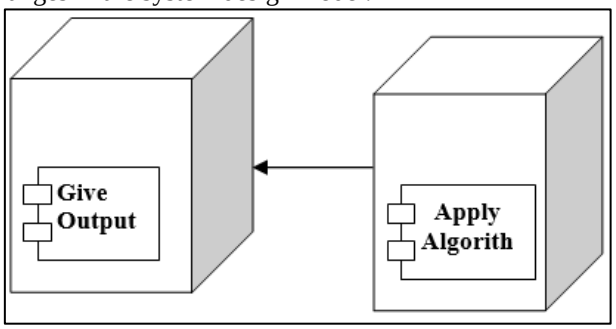

Fig. 1

The experimental data presented it can be concluded that the MAA algorithm is better than the Dyn FP-growth and FP-growth algorithm. First of all, the FP-growth algorithm needs at the most two scans of the database, while the number of

database scans for the candidate generation algorithm (Apriori) increases with the dimension of the candidate itemsets. Also, the performance of the FP-growth algorithm is not influenced by the support factor, while the performance of the Apriori algorithm decreases with the support factor.Thus, the candidate generating algorithms (derived from Apriori) behave well only for small databases (max. 50,000 transactions) with a large support factor (at least 30%). In other cases the algorithms without candidate generation Dyn FP-growth, FP growth and MAA algorithm behave much better .Several dataset have been used as test data to determine the performance and accuracy of the modified algorithm based on time and memory. The average results for both the execution time and the database pass yields 38% and 33% respectively in favor of the modified one. However, in some test data the outcome is in accordance with the original algorithm. It has been observed that as the number of items per transaction decreases the favorable result will be from the original algorithm since the pruning of candidate keys is closer to the first k+1 while implementing the modified one with the k(n) – 1 where n is the maxi-mum set size with set size frequency >= minimum support.

The sample data set has been generated for a number of items N = 100 and a maximum number of frequent itemsets |L| = 3000. |T| was chosen to be 10. Some of the results of the Comparison between the Apriori, MAA (Modified Apriori Algorithm), FP-growth and Dyn FP-growth algorithms for support factor of 5% and for different data sets are presented in Table. The execution time for the MAA algorithm is constant for a certain data set when the support factor decreases from 40% to 5% while, at the same time, the execution time of the MAA increases dramatically. For a support factor of 30% or greater and a data set of 40,000 transactions, the modified algorithm has better performances than the Apriori algorithm, but for a support factor of 20% or less its performance decreases dramatically. Thus, for a support factor of 5% the execution time for the Apriori algorithm is three times longer than the execution time of the FP-growth algorithm and up to five times longer than DynFP-growth.

Transactions(K)	Execution Time in sec			
	Apriori	DynFP-	FP-growth	MAA
10	13.94	2.32	3.76	2.06
20	21.98	3.98	6.88	3.12
30	48.37	8.23	14.63	6.10
40	66.50	12.10	20.90	11.26
50	107.65	19.50	34.30	15.36
80	198.30	37.90	64.80	33.89
110	1471.40	55.00	95.50	53.25
150	3097.20	98.90	174.60	86.85
190	5320.60	152.70	273.60	148355
300	9904.80	284.00	526.70	274.50
400	17259.20	458.10	849.70	258.25
520	20262.60	610.20		

Table 1

A. Time Comparison

The performance of modified algorithm has been compared with the FP growth, DnFP growth and Apriori. The run time is the time to mine the frequent itemsets. The experimental result of time is shown in Figure 5.6 reveals that the proposed scheme outperforms the modified Apriori approach. The graph shows that in terms of execution time, the modified apriori executes less time compared to the other mining algorithms. Moreover, in terms of database passes, the modified apriori provides less database access compared with the original one that makes its execution faster.

Fig. 2

B. Memory Comparison

The memory consumption for the modified algorithm has been producing high value at all level support because it produces candidate itemsets. The memory consumption for Dyn FP growth-at higher support levels is approximately compared to the new approach because as the support increase the probability of finding the maximal item set whose repetition is greater than the minimum support is less thus its working become same as the FPGrowth.

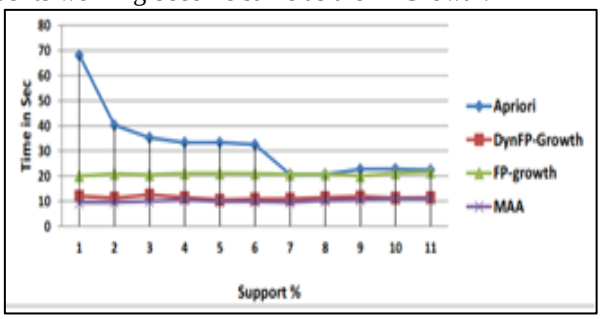

Fig. 3

Several experiments have performed to evaluate the performance modified one against FPgrowth, Dyn FP growth and Apriori, for generating the association rules. To perform the experiments different values of support were set because with different value of support the number of the frequent item sets is different, and the running time and the memory consumptions are affected by the value of the support From the experimental data presented it can be concluded that

modified Apriori algorithm (MAA) takes less time for generating frequent item and is efficient than other algorithms and it speeds up the data mining process.

IV. CONCLUSION

In this paper proposed new technique modified apriori algorithm provides in generating rules data mining. Technique in generating rules data mining. This algorithms is more efficient than the traditional algorithm and provide faster results in terms of time Time and memory complexity. The above comparison clearly states that new modifications in the Apriori can improve the efficiency of the apriori. The main attribute that always will be in consideration is the number of database scans. As the number of transaction grows the size of the database increases due to which the number of scans increases. Many algorithms above have suggested a new technique which requires only one database scan. These methods can also further be modified to increase the efficiency of apriori. Also candidate set generation is another important aspect that should be more focused on. The itemsets generation step of apriori many times generates item set which are not frequent and most of the time not required.

REFERENCES

[1] Won young Kim and Ungmo Kim, "Mining Frequent Itemsets with Normalized Weight in Continuous Data Streams", Journal of Information Processing Systems, Vol.6, No.1, March 2010.
[2] g.babu,dr t.bhuvaneswari "association rule mining for identifying optimal customers using maa algorithm" Journal of Information Processing Systems,Vol.66, No.3, 31st august2014.
[3] Kanimozhi Selvi Chenni angirivalsu Sadha sivam and Tamilarasi Angamuthu, "Mining Rare Item set with Automated Support Thresholds", Journal of Computer Science, Vol. 7, No. 3, pp. 394-399, 2011.
[4] Aghaebrahimi, Zahiri and Amiri, "DataMining Using Learning Automata",WorldAcademy of Science, Engineering andTechnology, Vol. 49, No.60, pp.308-311,2009.
[5] Mahesh, Mahesh T R and Vinayababu,"Using Data Mining Techniques forDetecting Terror-Related Activities on theWeb", Journal of Theoretical and AppliedInformation Technology, Vol.16, No.2, pp.99-104, June 2010.
[6] Bakır, Batmaz, Gunturkun, Ipekci, Koksal andOzdemirel, "Defect CauseModeling with Decision Tree andRegression Analysis", World Academy ofScience, Engineering and Technology,Vol.24, No.1, pp.1-4, 2006.

