Comparative Study of Association Rule Mining Algorithms

Gurneet Kaur1 Dr. R. K Chauhan2
1Scholar (M. tech) 2Professor
1, 2Computer Science and Applications Department
1, 2Kurukshetra University, Kurukshetra

Abstract--- Association Rule Mining (ARM) has been the area of interest for many researchers for a long time and continues to be the same. It is one of the important tasks of data mining. It aims at discovering relationships among various items in the database. There are several algorithms of which Apriori is the classical and most famous algorithm. In this paper, author considers two different datasets and tries to obtain the result using Weka a data mining tool. The author also presents a comparison between three association rule algorithms: Apriori Association Rule, Predictive Apriori Association Rule and Tertius Association Rule through experiments and presents the result. A proposal for increasing the efficiency of Apriori Algorithm is also presented.

Keywords: Association rule mining, Apriori, Weka.

I. INTRODUCTION

As Information Technology is growing databases used by the organizations are becoming large. These organizations include banking, manufacturing, marketing etc. To extract the valuable data, it is necessary to explore the databases efficiently. Data Mining helps to explore the valuable information from the database. It is the analysis step of the KDD(Knowledge Discovery and Data Mining) process. It is defined as the process of extracting interesting (non-trivial, implicit, previously unknown and useful) information or patterns from large information repositories such as: relational database, data warehouses etc. The goal of the data mining process is to extract information from a data set and transform it into an understandable structure for further use. Data mining has been given much attention in database communities due to its wide applicability. The problem of mining association rules from transactional database was introduced in [1]. The concept aims to find frequent patterns, interesting correlations, associations among sets of items in the transaction databases or other data repositories. Association rules are being used widely in various areas such as telecommunication networks, risk and market management, inventory control, medical diagnosis/drug testing etc.[3]

II. BASIC CONCEPTS

Association rule are the statements that find the relationship between data in any database. Association rule has two parts “Antecedent” and “Consequent”. For example {bread} => {eggs}. Here bread is the antecedent and egg is the consequent. Antecedent is the item that is found in the database, and consequent is the item that is found in combination with the first. A more formal definition can be given as [4]: Let I = {i1, i2, ..., in} be a set of items. Let D be a set of task relevant data transactions where each transaction T is a set of items such that T ⊆ I. A unique TID is associated with each transaction. Let A be a set of items. A transaction T is said to contain A if and only if A ⊆ T. An association rule is implication of the form A ⇒ B, where A ⊆ I, B ⊆ I, and A ∩ B = null. Association rule mining is done to find out association rules that satisfy the predefined minimum support and confidence from a given database. The problem of finding association rule is usually decomposed into two subproblems [7].

As shown in figure 1 one subproblem is to find those itemsets whose occurrences exceed a predefined threshold in the database, those itemsets are called large or frequent itemsets. The second subproblem is to generate association rules from those large itemsets with the constraints of minimal confidence. Suppose one of the large itemsets is Tk, Tk = {I1, I2, … , Ik}, association rules with this itemsets are generated in the following way: the first rule is {I1, I2, … , Ik-1} ⇒ {Ik}, by checking the confidence this rule is determined as interesting or not. Then the remaining rules are generated by deleting the last items in the antecedent and inserting them to the consequent, thereafter the confidences of the new rules are checked to determine their interestingness. This process is repeated until the antecedent becomes empty. Since the second sub problem is quite simple, most of the researchers focus on the first sub problem.

The first sub-problem can be further divided into two sub-problems: candidate large itemsets generation and frequent itemsets generation. The itemsets whose support exceed the support threshold are called as large or frequent itemsets and those itemsets that are expected or have the hope to be large or frequent are called candidate itemsets. The thresholds on which ARM technique are explained below:

- Support: is defined as the percentage/fraction of records that contain XUY to the total number of records in the database. It is given as: Support(XUY) = Support(XY)/Total number of transactions.
- Confidence: is defined as percentage of transaction in database containing X that also contain Y. It is given as: confidence = Support(XUY)/Support(X).
• Lift: is the ratio of the probability that X and Y occur together to the multiple of the two individual probabilities for X and Y. It is given as: \( \text{lift} = \frac{\text{Pr}(X,Y)}{\text{Pr}(X)\cdot\text{Pr}(Y)} \).
• Conviction: is similar to lift, but it measures the effect of the right-hand-side not being true. It inverts the ratio and is given as: \( \text{conviction} = \frac{\text{Pr}(X)\cdot\text{Pr}(\neg Y)}{\text{Pr}(X,Y)} \).

III. APRIORI ALGORITHM

Apriori is the most classical and famous algorithm for mining frequent patterns. It is a Latin word and it means “from what comes before”. It was introduced by [1]. The algorithm works on categorical attributes and employs bottom up strategy. It is based on the Apriori property which is useful for trimming irrelevant data. It states that any subset of frequent itemsets must be frequent.

![Diagram of the Apriori Algorithm](image)

**Fig. 2: Basic Framework to find Association Rules**

A. The Pseudo Code for Apriori Algorithm

The first pass of the Apriori Algorithm simply counts the item occurrences to determine large 1-itemsets. A relative pass, say k consists of two phases. Firstly, the large itemsets Lk-1 found in the (k-1)th pass are used to generate the candidate item-sets Ck. Next the support of candidates in Ck is counted. A hash-tree data structure can be used for fast counting.[2]

Discovering Large Item-sets
1) Pass 1
   i) Generate the candidate item-sets in C1
   ii) Save the frequent item-sets in L1
2) Pass k
   i) Generate the candidate item-sets in Ck from the frequent item-sets in Lk-1
   ii) Insert into Ck
   Select p.item1, p.item2, p.item3, ..., p.itemk-1, q.itemk-1 from Lk-1 p, Lk-1.q as follows:
   Insert into Ck
   Select p.item1, p.item2, p.item3, ..., p.itemk-1, q.itemk-1 from Lk-1 p, Lk-1.q
   Where, p.item1 = q.item1, ..., p.itemk-2 = q.itemk-2, p.itemk-1q.itemk-1
   • Generate all (k-1)-subsets from the candidate itemsets in Ck
   • Prune all candidate item-sets from Ck where, some (k-1) subset of the candidate item-set is not in the frequent item-set Lk-1
   ii) Scan the transaction database to determine the support for each candidate item-set in Ck
   iii) Save the frequent item-sets in Lk.

B. Limitations of Apriori

- Only the presence and absence of an item in transactional databases is explained.
- This algorithm is efficient only for small dataset.

- In Apriori, all items are treated equally.
- It requires large number of scans of dataset.
- The Minimum support threshold used is uniform. On the other hand other methods can address the problem of frequent pattern mining with non-uniform minimum support threshold [6].
- Apriori algorithm produces large number of candidate itemsets. So more time and resource are required in large number of scans [6].

C. Ways to Improve Apriori

- Transactions that does not contain frequent item-sets are of no importance in the next scans thus they can be removed.
- Hashing table can be used for counting the occurrences of item-sets.
- Partitioning: for any frequent item-set in database, then that item-set must be frequent in at least one of the partition of database [6].
- By adding attribute Weight and Quantity: implies how much quantity of item has been purchased.
- By adding attribute Profit: it can provide valuable information for business and customers.
- By reducing the number of scans on database.
- By removing the large candidates that lead to high Input/output cost.

IV. PREDICTIVE APRIORI ASSOCIATION RULE ALGORITHM

In this algorithm, support and confidence are combined into a single measure called “Predictive Accuracy”. In this association rule algorithm, this predictive accuracy is used to generate the Apriori association rule. It generates “n” best association rules based on the value specified by the user.[6]

V. TERTIUS ASSOCIATION RULE ALGORITHM

Tertius algorithm finds the rule according to the confirmation measures. It employs first order logic representation. It includes various options or parameters like class Index, classification, confirmation Threshold and values, frequency Threshold, missing Values, negation, noise Threshold, repeat Literals, roc Analysis, values Output, number literals etc.[5]. Only the first step consumes more time rest steps require less time.

VI. EXPERIMENTAL RESULTS

A. Working of Weka

Today, a large variety of data mining tool are available. Weka is one of the most widely used data mining tool. Basically, Weka is a collection of machine learning algorithms for the tasks related to data mining. It provides different functions like data preprocessing, clustering, classification and association rule mining etc. The algorithms can be applied directly on a dataset or through java code. The first step is to select the preprocess tab from where the dataset is selected using the open file option. After importing the data, the list of all the attributes is shown(Figure 3). Different options for applying filters on the database are present on the preprocess tab.
After importing the data, the associate tab is selected. From this tab the user can select the association rule mining algorithm of his or her own choice and their properties and perform the desired task (Figure 4).

**B. Database Scan for Apriori, Predictive Apriori and Tertius**

1) **Apriori ARM Algorithm:**
   Apriori algorithm uses the bottom up strategy and scans the database a number of times. The value of support used is 0.1 and the minimum confidence level is 0.9. The rules generated by Apriori algorithm are given in figure 5 below:

2) **Tertius Algorithm:**
   Tertius algorithm is based on hypothesis thus it also requires scanning database a number of times. It requires more amount of execution time. The rules generated by are given in figure 6.

3) **Predictive Apriori Algorithm:**
   It is based on accuracy (combination of support and confidence). It requires the maximum execution time to produce n rules specified by the user. Here the value of n used is 10. The rules generated by Predictive Apriori algorithm are given in Figure 7.

**C. Comparative Results**

The algorithms are compared on the basis of execution time they take by applying them on different number of instances of two different datasets. Table 1 shows the comparison of the three algorithms namely Apriori, Tertius and Predictive Apriori when they were applied on the first dataset which contains 11 attributes. The minimum support is 0.1 while the minimum confidence used is 0.9. The comparison has been made on the basis of their execution times on different number of instances of the dataset.
The comparison found out that Apriori is the fastest algorithm among these three algorithms. Its execution time does not depend on the size of the dataset while the execution time of other algorithms increases with the increase in the size of the dataset. The ascending order of the execution times of the algorithms is Apriori < Tertius < Predictive Apriori which shows that Predictive Apriori is the slowest of the three algorithms.

VII. PROPOSAL FOR AN IMPROVED APRIORI ALGORITHM

Apriori Algorithm is the classical algorithm for association rule mining. From the above experimental results it has been found that Apriori Algorithm gives best result among the three algorithms but still it has several drawbacks such as it requires multiple scan of the database. Moreover for candidate generation process it takes more memory, space and time. The rules generated by Apriori Algorithm consists of items which are irrelevant. This inefficiency of Apriori algorithm can be reduced by cutting down the number of transactions by deleting the transactions according to the size of the transaction and the value of K( kth pass). Moreover the efficiency can also be increased by reducing the redundant generation of sub-items during pruning of the candidate itemset Ck by further pruning Lk-1 before the generation of Ck. By applying both these measures the improved algorithm will be faster and thus increase the efficiency of data mining.

VIII. CONCLUSION AND FUTURE SCOPE

In this the author tried to find the best association rules using Weka ,a data mining tool. A brief introduction to all the three algorithms namely, Apriori, Tertius and Predictive Apriori has been given in the paper. In the second part author compared the association rule produced using the three association rule mining algorithms. The comparison among all the three algorithms has been made by applying them on different number of instances of two different dataset. The algorithms have been compared on the basis of their execution time. After comparing the author finds that Apriori algorithm is the fastest of them all.

Future Scope:
These algorithms can be used to bring out interesting relations among the data. Association rules produced by these three algorithms can be combined for better results. Algorithms can also be combined to form an efficient algorithm. The proposed algorithm can be implemented.

ACKNOWLEDGMENT

I am thankful to Prof R.K Chauhan, Dept. Of Computer Science and Applications, Kurukshetra University, Kurukshetra, India for his generous guidance and useful suggestions for this review work.

REFERENCES


All rights reserved by www.ijsrd.com 415


