Performance evaluation of Apriori algorithm using association rule mining technique

Jasleen Kaur\textsuperscript{a}, Rasbir Singh\textsuperscript{b}, Rupinder Kaur Gurm\textsuperscript{c}

\textsuperscript{a} MTech Student, jasleenkbhasin@yahoo.com, RIMT-IET, Mandigobindgarh, Punjab, India

\textsuperscript{b} Assistant Professor, rasbir.rai@gmail.com, RIMT-IET, Mandigobindgarh, Punjab, India

\textsuperscript{c} Assistant Professor, rupinderghurm@gmail.com, RIMT-IET, Mandigobindgarh, Punjab, India

Abstract

Many researchers studied the mining of data to get required information and knowledge in business applications, medical field, engineering projects and market data store. Discussing various data mining techniques and challenges posed by them. Association Rule mining proved to be best in many fields which generate relations between the items and generates strong rules using artificial intelligence with minimum confidence. These association rules is built into a large decision support system which can take managerial and operational decisions based on knowledge base developed and maintained by company. The most widely used Apriori algorithm for generating association rule discovers frequent patterns by candidate key generation which is a costly and memory consuming algorithm. Improvement in Apriori is ongoing topic these days. FP-Tree is another enhanced apriori algorithm which constructs frequent pattern tree without candidate generation. This makes it cost effective and more efficient. Other algorithms like K-means, Hash tree, bayesian networks, neural networks are discussed briefly. The proposed work based on making Apriori less complex by transposition techniques. Further improvement can be done by altering support count formula and reducing the number of transactions.

Keywords— Data mining, Apriori algorithm, Association Rule Mining, FP-Tree algorithm, efficiency

I. INTRODUCTION

Data mining is the concept having a great scope of research for scientists. Data mining deals with different kinds of data from large databases which undergoes various processes to get information and knowledge base is created [7]. Collecting and storing of large databases is necessary for online services whether it is social media services or online shopping websites. Data can be mined using various techniques depending upon kind of data to be mined. Earlier, researchers proved that by collecting and analysing historical data, we can well predict the future values which are beneficial in any field to increase the business opportunities. Data mining can be used in answering the queries, to manage large knowledge bases or as a decision support system.

Data can be of any form, mostly available as relational data. E.g. in audio format (.mp3 or .wav), video format (.mp4), documents, files, number, string, etc. this constitute complex data. One may invent powerful
system to mine all kinds of relational, complex, transactional data simultaneously. The key parameters to determine complexity of data mining algorithms are scalability, efficiency and running time. Algorithm with low complexity is of practical use. The sensitive data should be handled carefully. The knowledge obtained from data should in user’s understandable form, otherwise it is complex process. So system should contain knowledge discovery and representation system within itself. Data being shared in multi-level mining should be kept secured and protective. These security goals are contradicting to multi-level mining. Big data is the emerging trend in data mining which poses the challenge of managing large volumes of data with increasing speed of networks and use of World Wide Web [17]. Sentiment analysis is the new technology where words are recorded to analyze user sentiments [12].

A. Challenges of data mining

1) Clustering:
Clustering is the process of discovering the clusters or the patterns [8]. The items belonging to particular cluster may share common properties. To search the space of clusters, control strategy is used which should clearly define boundaries of clusters and should not be expensive. Iterative optimisation techniques are used to implement clustering. Post clustering analysis, re-sampling based pruning strategies are used to superfine the clusters repeatedly in iterations. Clustering is used to discover knowledge or for simply partitioning purpose. There are two main cluster techniques: Partitioning and hierarchal clustering.

(i) Hierarchal clustering:
It divides items into groups of having common parent. It is implemented using structure tree model. It is used to organise the information E.g in the web portal. Clusters are made using top-down or bottom-up approach. Top-down approach (divisive) starting with single cluster divided into smaller clusters while choosing best decision. Bottom-up approach (agglomerative) starts with single cluster merges with other cluster to form bigger one until all the clusters are united.

(ii) Partitional clustering:
Each instance is put into one of the non-overlapping cluster based on some predefined property. Based on partitioning, K-means is applied on data. This method initialises k, number of clusters which should be known before. It places each instance into clusters according to calculated means of each cluster centres. Other method used for partitioning is Gaussian Mixture models (GMM) gives graphical models of each clusters consisting of their respective members with known value of k, number of clusters.

The new image processing technique based on partitioning set in hierarchal trees is very fast and effective [11]. Image compression includes coding and decoding (codec) algorithm.

2) Association Rule mining:
Association Rule mining discovers frequent patterns of the data sets obtained from large repositories [10]. The formal model of association rule was given by R. Agrawal [2]. Let, \(A = a_1, a_2, \ldots, a_m\) be a set of \(m\) different item set and \(T\) be the transactions which comprises of item set from set \(A\) such that \(T \subseteq \mathcal{D}\) is the database with transactions \(T\). This rule mining model can be expressed as \(P \Rightarrow Q\), where \(P, Q\) are the item sets and \(P \subseteq Q = \emptyset\). \(P\) is an antecedent whereas \(Q\) is a consequent. This is known as \(P\) implies \(Q\). The two thresholds used in association are minimal support and minimal confidence. Support is measure of existence of \(P \subseteq Q\), proportions of records containing both \(P\) and \(Q\) from database \(D\).

\[
\text{Support} (P, Q) = \frac{\text{Support sum of } P \cap Q}{\text{Overall records in } D}
\]

Confidence is the ratio of transactions containing both \(P\) and \(Q\) to that of Transactions containing \(Q\) only.

\[
\text{Confidence} (P|Q) = \frac{\text{Support} (PQ)}{\text{Support} (P)}
\]

Multilevel association rule mining is applied to large databases using top-down approach [5]. Data is abstracted at multiple levels. Strong negative rules can be explained as a consequence of surprising patterns such as Penguin=> flies. The rule like \(M \Rightarrow N\) is positive while rules with negation sign, \(M \Rightarrow \neg N\), \(\neg M \Rightarrow N\) and \(\neg M \Rightarrow \neg N\) [13].

3) Prediction analysis:
Prediction technique stores and analyzes historical data for prediction of future data. Data mining contains hidden relationship in data[18]. Three main activities are involved in prediction model. First, is the discovery of pattern. At last the forensic testing is done to determine any abnormal pattern. Then actual prediction function is applied to predict future value. This method is used to predict faults in the grid environment which keeps on changing dynamically so as to avoid the failures or disruption in workflow activities [9].

Neural Networks, fuzzy logic, genetic algorithms and rough set theory are the techniques used to predict future values. 100 percent accuracy is not guaranteed. It is difficult to build a probability model for real time industries.

(i) **Neural Networks**: Neural network is defined as system with highly inter-connected processing elements working on dynamic input to give response in form of output. There are further two neural network. One with feed-forward network, having no feedback loop and is uni-directional and other is feed-back mechanism which gives feedback path.

(ii) **Bayesian networks**: It is graphical structure consisting of random variable nodes and its associated probabilities.

### B. Association Rule Mining algorithms:

Based on the concept of association rule mining there are basically two main algorithms: Apriori and Eclat algorithm. Support is the non-quantitative measure whereas confidence is measure of how strong the association rule is [19]. Use of Apriori is now not limited to market basket data but also in medical field billing, educational data [1], World Wide Web data, etc. The fast association algorithms are efficient having least running time and high scalability [15]. The problem of generalised association rule is taken up by Ramakrishnan and R. Agrawal during the IBM research [16].

#### 1) *Apriori Algorithm*:

Apriori Algorithm is the algorithm to find frequent item sets existing in databases with multiple scanning of data. From these frequent item sets, the strong association rules are generated. R. Agrawal in 1993 discovered the frequent item set generation algorithms for increasing speed of mining.

Basic Apriori algorithm contains commonly of two steps join and prune actions.

(i) **Join action**: Let $k$ is the item sets present in set $L_k$, which is a frequent set. To find $L_k$, join operation is performed between $L_{k-1}$ with itself.

(ii) **Prune action**: Let, $C_k$ be candidate set which is superset of $L_k$, then items from $C_k$ according to Apriori are removed from $L_k$ if having value less than threshold.

**Steps for Apriori algorithm are:**

1. **Step 1**: Set the user predefined minimum support and confidence.
2. **Step 2**: Construct first candidate set and name it as $C_1(k-1)$ having item sets $C_1, C_2, \ldots, C_n$. Now perform prune operation by removing item sets with support values lower than threshold. Here, frequent-1 item set($L_1$) is obtained.
3. **Step 3**: Join $L_i$ with itself to obtain $C_2$, candidate item sets- 2($k$). Again remove infrequent item sets from $C_2$ to get $L_2$, frequent item set-2.
4. **Step 4**: Keep repeating the step 3 until no more candidate set is generated.

The **pseudo code** for apriori is written.

$C_k$: Candidate itemset of size $k$

$L_k$: frequent itemset of size $k$

$L_i = \{\text{frequent items}\};$

for $(k = 1; L_k \neq \emptyset; k++)$ do begin

$C_{k+1} =$ candidates generated from $L_k$;

end
for each transaction \( t \) in database do
  increment the count of all candidates in \( C_{k+1} \) that are contained in \( t \)
  \( L_{k+1} = \) candidates in \( C_{k+1} \) with min_support
end

return \( \cup_k L_k \);

Further, improvement in apriori is done by minimising the candidate generation keys[6]. To optimise space and time complexity, the algorithms without candidate generation are introduced such as hash tree mapping or frequent fragment tree algorithm[14]. Zang Li extended the basic Apriori algorithm to minimise its limitations [3]. The problem in Apriori is when support is set low, then frequent sets and rules are produced large. As the result, association rule thus formed has larger confidence but low support. Concept of multiple support value association rule was discovered. To correct, two types of extended Apriori algorithm, are generated. One is Apriori-con algorithm based on confidence. Second is based on classification. Apriori was further improved by decreasing pruning operations of candidate item set. Methods used to improve Apriori like hash based technique, sampling and vertical data format [4]. Partitioning also reduces CPU time and effort. The algorithm calculates support count similar to Apriori and discovers frequent item sets. Then generate candidate-2 item set and calculate their support count. It deletes infrequent itemsets from transaction with which it is associated and into main memory, thus reducing memory space. New technique, divides data set into different horizontal partitions.

2) Eclat algorithm:
Eclat like other association rule works on the idea that calculation of support of candidate item set is done using tid set intersections so as no other subsets which are not present in prefix tree are generated. Data can be recorded in form of pair(A, t(A)), t(A) is tid set of item A. Initially all single items with their respective tidsets are taken. Then recursively the function IntersectTidsets is applied to all itemset-tidset pairs (A, t(A)) with all other pairs (B, t(B)) to generate new candidate set \( P_{A,B} \). If items are frequent then they are added to \( F_A \). Likewise, recursively frequent item sets are generated.

II. ANALYSIS OF EFFECTIVENESS OF APRIORI AND FREQUENT PATTERN TREE ALGORITHM
From several years, IT professionals are facing difficulties while handling of software risks in the projects. On the other hand the software companies have to meet the deadlines, otherwise it may result into worst scenario. Software engineering is the process which deals with demarcation, evaluation, and recovery of the software development life cycles itself. Software risks occur at different phase of development of software. There are such mitigation factors which can help to avoid software risks. A recent research done by Asif [20] has proposed the new rule based decision support system generating rule by artificial intelligence concept. This helps to understand relations between risk factor and its associated mitigations. Another advancement is done to improve Apriori algorithm named as Frequent Tree (FP-Tree) algorithm. This research targeted upon working of Apriori and FP-Tree algorithm on the risk factors and mitigation dataset. Comparison of advantages and disadvantages of these two algorithms. At last, the embedded architecture is developed with mining techniques.

TABLE I. MAIN DATASET
Two-dimensional dataset array has list of software risks and its mitigation factors from [20]. Abbreviations can be noted from [20].

**A. Tracing of Apriori algorithm:**

**Step 1:** To calculate minimum support, candidate and frequent item sets generation.

Asif assumed minimum support (S) of 15% and by applying the formula of support we get value of 0.3.

\[
\frac{15}{100} \times 20 = 0.3
\]

Now fixing threshold as 0.3, we will start generating candidate items by stepwise Apriori algorithm already discussed.

Output: Frequent items = (M1 M3)(M1 M12)(M2 M3)(M21 M22)

**Step 2:** To generate rules with minimum confidence.

Minimum confidence (C) is assumed to be 60%. Formula of confidence is \( C(P,Q) = \frac{S(\text{PUQ})}{S(P)} \). For all four frequent item set pair, confidence is calculated and confidence of each rule is determined.

**B. Tracing of frequent pattern tree algorithm**

This generates frequent patterns without generation of candidate items. It generates frequent tree.

**Step 1:** Calculate minimum support (15%).

**Step 2:** Find the number of times or frequency, the mitigation item exists.

**Step 3:** Select the items which has equal to or more than minimum support which is 0.3 and remove the remaining.

**Step 4:** Then order the items prioritywise like item with highest frequency is placed first.

**Step 5:** Draw frequent pattern tree according to number of counts of each item.

**Step 6:** Frequent pattern is generated by recursively applying three steps: (i) conditional pattern base. (ii) conditional FP-Tree (iii) Recursive mining of conditional FP-Tree and generate frequent patterns.

**C. Comparison of Apriori and FP-Tree algorithms and conclusions**

With FP-Tree algorithm database scan has reduced from three to two times without any candidate generation which is highly hectic. Database is kept as small as possible where each node of tree store a counter in form of linked list. Further divide and conquer method used in FP-Tree algorithm make tasks simpler and makes it more memory efficient as comparison to Apriori. Decision Support system thus developed based on Apriori and FP-Tree makes it more adaptive in nature. There is always scope for improvement in increasing efficiency of Apriori algorithm.

**III. PROBLEM FORMULATION**

In association rule mining major problem was to find the different items associations in data sets. The Apriori algorithm is the most efficient algorithm to generate these association rules. But in apriori algorithm
for finding the frequent item sets repeatedly database scan is done which takes much time and degrades the performance of apriori algorithm. The apriori algorithm is mostly applied on many applications to provide the rules generated for that specific application and also give the details about items.

In apriori algorithm is applied for large datasets and step wise search is used. Apriori property helps to search the frequent items and rules which are mostly used. Many database scans are necessary in this algorithm when the database size is large. This technique will much time taken and need to be improved. The efficiency of Apriori algorithm will be enhanced by reducing the transitions which further reduce the time need to be consumed to generate the association rules. So, in this work, proposed technique will work on reducing the number of transitions of apriori algorithm. It will also reduce the similar sub-items generation during prune step and also candidate having not frequent sub items are deleted. This will provide association rules generated with low load on available resources and also in less time.

IV. EXPECTED OUTCOMES

Using proposed technique the number of database scans will reduce. As the total scans over database reduced, the time taken by algorithm will also decrease. Proposed technique will reduce the escape time of Apriori’s algorithm for association rule generation. The number of steps required to generate the frequent item sets will be reduced, the complexity of Apriori algorithm will be reduced in some manner.

V. CONCLUSION

Association rule mining is one of the important techniques of research used in the data mining. For generating association rules, the Apriori algorithm is considered as the most efficient one. Apriori algorithm with a different technique is discussed which will acquire enhancement in Apriori algorithm with the transposition of database. Further improvement will be done in transposition technique using some different calculations of minimum support count. This approach will reduce the total scans over database as well as take less time to generate the association rule.

REFERENCES


