Abstract—Frequent item set mining is a heart favorite topic of research for many researchers over the years. It is the basis for association rule mining. The concept of frequent item set mining together with the pros and cons of the current methods have been elaborated. Frequent item set mining is a very popular and computationally expensive task. In this paper, we have developed a method to discover large item sets from the transaction database. The infrequent elements have been pruned from the source data base. The proposed mining method is efficient in terms of the time and space consumption.

Key words: Data Mining, Frequent Item Set, Association Rule

I. INTRODUCTION

In the previous decade or so the size of data base has increased exponentially. This has led to a growing interest in the development of tools capable in the automatic extraction of knowledge from data. The data mining or knowledge discovery in data base is related to finding useful patterns from a large amount of data. All the discovered trivial information is used for the prediction & generally known as patterns. Of all the mining functions in the knowledge discovering process, frequent pattern mining is to find out the frequently occurred patterns. The measure of frequent patterns is a user-specified threshold that indicates the minimum occurring frequency of the pattern. We may categorize recent studies in frequent pattern mining into the discovery of association rules and the discovery of sequential patterns. Association discovery finds closely correlated sets so that the presence of some elements in a frequent set will imply the presence of the remaining elements (in the same set).

The KDD process is shown below in Fig. 1

Fig. 1: The process of knowledge discovery in databases [2]

Data mining is the discovery of hidden information found in databases and can be viewed as a step in the knowledge discovery process. Data mining functions include clustering, classification, prediction, and link analysis (associations). One of the most important data mining applications is that of mining association rules. Association rules are first introduced by Agarwal. Association rules are helpful for analyzing customer behavior in retail trade, banking system etc. Association rule can be defined as \( X, Y \implies Z \). In retail stores if customer buys \( X, Y \) he is likely to by \( Z \). Concept of association rule today used in many application areas like intrusion detection, biometrics, production planning etc. Association rule mining is defined as to find out association rules that satisfy the predefined minimum support and confidence from a given data base. If an item set is said to be frequent, that item set supports the minimum support and confidence. The problem of finding the association rules can be divided into two parts:

1) Find all frequent item sets: Frequent item sets will occur at least as frequently as a pre-determined minimum support count i.e. they must satisfy the minimum support.

2) Generate strong association rules from the frequent item sets: These rules must satisfy minimum support and minimum confidence values.

Frequent pattern mining is the process of mining data in a set of items or some patterns from a large database. The resulted frequent set data supports the minimum support threshold. A frequent pattern is a pattern that occurs frequently in a dataset. A frequent item set should appear in all the transaction of that data base.

A. Proposed Technique:

The proposed algorithm is as follows:

1) Inputs:
   - Transaction data bases \( D_1, D_2, \ldots, D_n \)
   - MST – Minimum Support Threshold
   - MCT – Minimum Confidence Threshold

2) Output:
   - A Set of Association Rules

3) Procedure:
   a) Phase 1: combine the databases \( D_1, D_2, \ldots, D_n \) in to a single data base \( D \). It is the union of \( D_1, D_2, \ldots, D_n \)
   b) Phase 2: Frequent Item Set Mining
      1) Step 1: Scan the transaction data base & find the support count of each item
      2) Step 2: Eliminate the infrequent items
      3) Step 3: Now arrange the frequent items in the decreasing order of their support count. This order will be used in the construction of the compact tree ( CF-Tree)
      4) Step 4: Construct CF-Tree by reading 1 transaction at a time
      5) Step 5: Extract a sub tree ending in an item (For example, if \( e \) is the last item in a transaction database than we have to find a sub tree ending in \( e \))
6) Step 6:
   - Check that the item of step 5 (i.e. e) is frequent or not
   - If it is frequent then extract it as frequent item
   - New item (e) is frequent so now find the other frequent items ending with (i.e. be,de,ce,...)
   - Continue this recursive procedure until no item found

Phase 3: association rule mining
For each frequent item set X
For each non empty subset A of X
Let B = X – A
Then A -> B is an association rule if the confidence of (A -> B) >= minconf

B. Result Analysis:
1) Input Data Set
The input data set is as follows [5]:
   1 3 4
   2 3 5
   1 2 3 5
   2 5
   1 2 3 5
The MST is 40%.

C. Previous Algorithm:
The results of the algorithm are as follows:
1) Output:

<table>
<thead>
<tr>
<th>Item set</th>
<th>Support level</th>
<th>Item set</th>
<th>Support level</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>2,5</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>3,5</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>1,2,3</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>1,2,5</td>
<td>2</td>
</tr>
<tr>
<td>1,2</td>
<td>2</td>
<td>1,3,5</td>
<td>2</td>
</tr>
<tr>
<td>1,3</td>
<td>3</td>
<td>2,3,5</td>
<td>3</td>
</tr>
<tr>
<td>1,5</td>
<td>2</td>
<td>1,2,3,5</td>
<td>2</td>
</tr>
<tr>
<td>2,3</td>
<td>3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Output
Candidates count: 15
Algorithm stopped at size 5, because there is no candidate
Frequent itemsets count: 15
Maximum memory usage: 0.55 mb
Total time ~ 40 m

D. Proposed Algorithm:
The results of the algorithm are as follows

<table>
<thead>
<tr>
<th>Item set</th>
<th>Support level</th>
<th>Item set</th>
<th>Support level</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>2,1</td>
<td>2</td>
<td>3,2</td>
<td>3</td>
</tr>
<tr>
<td>3,1,2</td>
<td>2</td>
<td>5,2,3</td>
<td>3</td>
</tr>
<tr>
<td>5,1,2,3</td>
<td>2</td>
<td>5,2</td>
<td>4</td>
</tr>
<tr>
<td>5,1,2</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>3,1</td>
<td>3</td>
<td>5,3</td>
<td>3</td>
</tr>
<tr>
<td>5,1,3</td>
<td>2</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>5,1</td>
<td>2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2: Results of the algorithm
Number of frequent itemsets: 15
Total time ~: 25 ms
Max memory: 0.5078125

II. Conclusion
In this paper, we presented a novel algorithm for mining frequent item sets. Frequent item set mining is crucial for association rule mining. The proposed technique is eliminating the infrequent items from the source data set to generate a compact data set. We have evaluated the performance of our proposed algorithm. It is fast. Also it is taking less main memory for computation in comparison to previous algorithm.
REFERENCES