Evaluation of Frequent Itemset Mining Algorithms - Apriori and FP Growth

Jismy Joseph, Dr. G. Kesavaraj

1PhD Research Scholar, 2Professor and head
Department of Computer Science, Vivekanandha College of Arts and Science for Women (Autonomous), Elayampalayam, Thiruchengode, Tamil Nadu, India

Abstract- Nowadays the Frequent Itemset mining (FIM) is an essential task for retrieving frequently occurring patterns, correlation, events or association in a transactional database. Understanding of such frequent patterns helps to take substantial decisions in decisive situations. Multiple algorithms are proposed for finding such patterns, however the time and space complexity of these algorithms rapidly increases with number of items in a dataset. So it is necessary to analyze the efficiency of these algorithms by using different datasets. The aim of this paper is to evaluate the performance of frequent itemset mining algorithms, Apriori and Frequent Pattern (FP) growth by comparing their features. This study shows that the FP-growth algorithm is more efficient than the Apriori algorithm for generating rules and frequent pattern mining.

Keywords- Frequent itemset mining, Apriorialgorithm, FP-growth algorithm, Weka, Association Rule.

I. INTRODUCTION

These days the size of databases increases rapidly, this leads to invention of different tools to extract information automatically from the large database. Many researchers used data mining or knowledge discovery in database (KDD) to extract innovative and frequent pattern from large databases or transactional databases.

Frequent itemset mining has been applied in a great number of fields, including intrusion detection, Market basket analysis and credit card fraud prevention to discover unpredicted association among itemset in transactional and relational database. Frequent pattern Mining or association mining shows which items appear together in a transaction or relation.

Frequent patterns are patterns such as subsequences, itemsets or substructures that appears in a data set frequently. Finding such frequent pattern plays an essential role in mining association, correlations, and many other relationships among the data. It also helps in data classification, clustering and other data mining tasks as well. Thus, frequent pattern mining has becomes an important data mining task and a focused theme in data mining research [1].

II. LITERATURE REVIEW

Much research work has been done to compare different algorithms used for frequent itemset. In the paper ‘Survey on Frequent Item set Mining Algorithms’ [2] Pramod S and O.P. Vyas conducted a survey on frequent itemset mining algorithms. They used Adult and Mushroom dataset for evaluating the performance of different algorithm. This study comprises features like different support values, size of transactions and different datasets. They concluded that SaM algorithm performed better in all data set.

In [3] M.S Mythili, A.R Mohamed Shanavas conducted a Performance Evaluation of Apriori and FP-Growth Algorithms by comparing the capabilities of these algorithms. The study show that FP-growth algorithm is more efficient than the Apriori algorithm.

In [4] Mr. Rahul Shukla, Dr. Anil kumar Solanki conducted a performance Evaluation for Frequent Pattern mining Algorithm. In the paper they compared the performance of Apriori and ECLAT Algorithm on medical data and they concluded that the Eclat approach is more efficient for mining frequent patterns in a large database.
In the paper ‘Algorithms for frequent itemset mining: a literature review’ Chin-Hoong Chee, Jafreezal Jaafar, Izzatdin Abdul Aziz, Mohd Hilmi Hasan & William Yeoh reviewed the strengths and weaknesses of the important and recent algorithms in Frequent Pattern Mining and they identified two major problems with frequent pattern mining. They mentioned that frequent hidden pattern mining needs more time and memory [5].

In the paper[6] Frequent Itemsets Mining for Big Data: A Comparative Analysis Daniele Apiletti, Elena Baralis, Tania Cerquitelli, Paolo Garza, Fabio Pulvirenti and Luca Venturini conducted a theoretical and experimental comparative analyses of Hadoop- and Spark-based scalable algorithms to find frequent pattern from big data.

In [7] Ravi Ranjan and Aditi Sharma compared Hadoop, Spark, Flink by using Apriori and FP-Growth on different dataset. They suggested that Flink is performing better in the field of big data.

III. FREQUENT PATTERN MINING (FPM) AND ALGORITHMS

In data mining tasks, the frequent item sets play an important role for finding frequent pattern or association from various kinds of databases like relational database, transactional database etc. FPM plays an essential role for clustering, classifying and identifying outliers a set of data. Apart from this, FPM has many applications like spatiotemporal data analysis, biological data analysis, and software bug detection [8].

FPM is used to predict the occurrence of a specific item based on the occurrence of other items in the transaction. The terminologies used in frequent pattern mining are support and confidence. Support specifies how frequently an itemset appears in the dataset and the confidence describes how often the rule has been found to be true. The support and confidence are defined as:

\[
\text{Support } (A \Rightarrow B) = P(\text{A U B}) \\
\text{Confidence } (A \Rightarrow B) = \frac{\text{Support } (A \cup B)}{\text{Support } (A)}
\]

The rule is considered as strong if it satisfies a minimum support threshold and a minimum confidence threshold. The methods used in frequent item set mining are

- Finding Frequent itemsets using candidate generation
- Mining Frequent itemsets without candidate generation
- Mining Frequent itemsets using Vertical Data format.

APRIORI Algorithm

R. Ararwal and R. Srikant proposed this algorithm in 1994 for mining frequent itemsets for Boolean association rule [1]. It uses candidate generation for finding frequent itemsets. Aprioris an iterative approach, where k-itemsets are used to find (k=1)-itemsets. This algorithm uses Apriori property to reduce the search space. It is a two-step process called join and prune.

Join step - To find L_k, a set of k-itemset is generated by joining L_{k-1} with itself.

Prune step - Scans the count of each item. If it less than minimum support then it does not considered as frequent.

Apriori Algorithm Pseudocode

Procedure Apriori(D, minSupport) // D – database, minsupport- Minimum support
{
    L1 = \{frequent items\};
    For (k= 2; Lk-1! =\emptyset; k++)
    {
        C_k= candidates generated from L_{k-1}
        For each transaction t in database do
        {
            Increment all candidates
            L_k = candidates in C_k with minSupport
        }
    } Return C_k L_k;
}

B. FP-Growth Algorithm

The main disadvantages of Apriori algorithm are it generates huge number of candidate sets and this algorithm repeatedly scans the database and checks a large number of candidates by pattern matching. Hence it is very costly. To overcome these
disadvantages, the next method for generating frequent itemset without using candidate generation is FP-Growth. FP-Growth uses a divide and conquer approach. It first compresses the database into FP-Tree and then divides these compressed database into a set of conditional database.

**FP-Growth Algorithm**

Step1: Scan the database to get set of frequent itemset and their support count.
Step2: Sort the frequent itemset in descending order using support count.
Step3: Construct FP-tree. Initially it creates the root of the tree and labelled as ‘Null’.
Step4: Construct the FP-conditional tree for each item (or itemset)
Step5: Determine the frequent patterns.

**IV. DATASETS AND RESULTS**

We have used supermarket data set and vote datasets from ‘storm.cis.fordham.edu’ for comparing two frequent pattern mining algorithms. The first data set is ‘Super Market Data Set (SMDS)’ [9], which contains 4627 instances and 217 attributes. The second one is ‘Vote Data Set (VDS)’ [9], which consists of 435 instances and 17 attributes. In this comparative study, two frequent itemset mining algorithms are used. They are Apriori and FP growth.

Initially, the datasets are preprocessed and afterwards the algorithms are applied. The following figures show the results obtained from weka after preprocessing. Fig.1 shows the result obtained after preprocessing the dataset ‘SMDS’.

![Fig.1. Preprocessing Result – SMDS](image1)

Fig.2 shows the result obtained after preprocessing the dataset ‘VDS’.

![Fig.2. Preprocessing Result – VDS](image2)

The following figures show the output obtained from Apriori and FP growth algorithms. Fig.3 shows the rules generated by Apriori algorithm in SMDC dataset. Ten rules are generated based on support and confidence.

![Fig.3. The Rules Generated by Apriori Algorithm in SMDC Dataset](image3)

Fig.4 shows the rules generated by Apriori algorithm in VDS. Ten rules are generated based on support and confidence.

![Fig.4. The Rules Generated by Apriori Algorithm in VDS](image4)

The rule generated by FP-Growth using vote data set is shown in Fig 5. FP Growth found 41 rules in vote dataset and only the top 10 rules are displayed in the following figure.

![Fig.5. The Rules Generated by FP-Growth Algorithm in VDS](image5)
Fig 6 shows the rules generated by FP Growth in SMDC data set. FP Growth found 16 rules in vote data set and only the top 10 rules are displayed in the following figure.

VI. COMPARISON OF RESULTS

This section deals the comparison part of Apriori and FP Growth algorithms. The table 1 shows the time needed to generates rules by using Apriori and FP Growth. Compared to Apriori the FP Growth algorithm takes less time to generate the rules in both datasets. In the supermarket dataset, Apriori algorithm took 4 seconds to produce the rules when minimum support is 0.15. But FP growth algorithm generated the rules within in a second. In the case of vote dataset FP Growth performed better than the Apriori algorithm.

<table>
<thead>
<tr>
<th>Instance</th>
<th>Attribute</th>
<th>Minimum Support</th>
<th>Time Taken (in Sec)</th>
<th>Apriori</th>
<th>FP Growth</th>
</tr>
</thead>
<tbody>
<tr>
<td>SupperMarket</td>
<td>217</td>
<td>0.15</td>
<td>4</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.2</td>
<td>12</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Vote(435)</td>
<td>17</td>
<td>0.15</td>
<td>2</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.2</td>
<td>2</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

TABLE 1- COMPARISON RESULTS

VI. CONCLUSION

In many data mining applications, association rule plays an important role for finding frequent pattern. In this study we observed that FP Growth algorithm is better than the Apriori algorithm. In both datasets the FP growth taken less time to generate the rule. FP-growth is more acceptable for larger databases.

References