

**Project Report**

**Project Name:** FP Growth and Apriori algorithm for data mining

Course Code: CSE477

Course Title: Data Mining

Section: 02

Summer-2021

**Submitted By:**

Sanjida Akter

ID: 2018-1-60-089

Samia Sultana

ID: 2017-2-60-122

Md. Nadim

ID: 2018-1-60-161

Md. Ibrahim Khalil Mehedi

ID: 2018-1-60-152

Md Mizanur Rahman

ID: 2018-1-60-249

**Submitted To:**

Jesan Ahammed Ovi

Senior lectuter

Department of Computer Science and Engineering

**Introduction**: In Data Mining, Association Rule Mining is a standard and well researched technique for locating fascinating relations between variables in large databases. Association rule is used as a precursor to different Data Mining techniques like classification, clustering and prediction. To measure the performance of the Apriori algorithm and Frequent Pattern (FP) growth algorithm by comparing their capabilities in different datasets. The evaluation study shows that the FP-growth algorithm is efficient and ascendable than the Apriori algorithm has many differentiative approach for serving reslts.

Here, we are analysing chess and mushroom datasets in terms of apriori and fp growth algorithm.

**Assosiation rule mining** : Association rule mining is a procedure which is meant to find frequent patterns, correlations, associations, or causal structures from data sets found in various kinds of databases such as relational databases, transactional databases, and other forms of data repositories. An Association rule is an expression of the form A → B means that whenever A seems, B also tends to appear. A and B are itemsets. An itemsets is nothing but a collection of database items. A is usually stated as the rule’s antecedent and B as the consequent of the rule. It finds all the frequent itmsets and generates strong assosiation rules from frequent itemsets.

**Apriori algorithm for data mining:** For mining data all non-empty subset of frequent itemset must be frequent. The key concept of Apriori algorithm is its anti-monotonicity of support measure. Apriori property is,

All subsets of a frequent itemset must be frequent. If an itemset is infrequent, all its supersets will be infrequent*.*

1. Determines the support of itemsets given in the transactional database, and select the minimum support and confidence.
2. Take all supports in the transaction with higher support value than the minimum or selected support value.
3. Find all the rules of these subsets that have higher confidence value than the threshold or minimum confidence.
4. Sort the rules as the decreasing order of header table.

**Major drawbacks of Apriori algorithm:** The algorithm scans the datasets several times thus, n-length frequent pattern creates n-length data scanning. It results in storing unnecessary candidate data which is resourse consuming.

**FP Growth algorithm:** FP growth algorithm is an efficient algorithm for producing the frequent itemsets without generation of candidate itemsets. It adopts a divide and conquer strategy and it needs two database scans to seek out the Support count. It can mine the items by using lift, leverage and conviction by specifying minimum threshold. Steps for the algorithm,

1. Database first scan to find frequent single itemset pattern.
2. Construct header table by Sorting frequent items in frequency descending order.
3. Database 2nd scan to construct fp tree.
4. Constructing the conditional FP tree in the sequence of reverse order header table to generate frequent item set.

**Comparing Apriori and FP growth algorithm:**

|  |  |
| --- | --- |
| **Apriori algorithm** | **FP-growth algorithm** |
| Array based structure | Tree based structure |
| BFS search type | Divide and concure technique |
| K+1 number of database scan | 2 database scan |
| Large memory required | Less memory required |
| Runtime is more | Runtime is less |
| For sparse datasets | For large and medium datasets |

**Datasets analysis:** The data has to be handled efficiently to get the best outcome from the Data Mining process**.** We are analysing chess and mushroom datasets.

|  |  |  |
| --- | --- | --- |
| **Dataset name** | **Number of data** | **Number of attributes** |
| Chess | 3196 | 37 |
| Mushroom | 8124 | 23 |

***For chess datasets***

The threshold value:

[0.6, 0.65, 0.7, 0.75, 0.8, 0.85, 0.9, 0.95]

Runtime for fp-growth:

[17.6,7.5,3,2.3,2.1,2,1.6,1.2]

Runtime for apriori:

[90,25,15,13,8,5,3,2]

***For mushroom datasets***

The threshold value:

[0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0]

Runtime for fp-growth:

[14.2, 0.81, 0.22, 0.18, 0.16, 0.15, 0.14,0.13, 0.12]

Runtime for apriori:

[20,10,2,1.9,1.7,1.6,1.5,1.4,1.2]

**Result analysis:** After observing the chess and mushroom datasets in both apriori and fp-growth algorithm, fp-growth shows a better result than apriori algorithm.

Here, first we check the minimum threshold support against runtime for the datasets chess and mushroom. Because of the tree structure fp-growth algorithm uses less memory and works faster. It also scans the datasets 2 times for mining process rather than taking k-1 times in apriori algoritm. It is efficient and scalable for mining both long and short frequent patterns.

Apriori algorithm is a fundamental aproach to data mining but fp-growth is definitely an improvement for mining process.

**Runtime with different minimum support:**

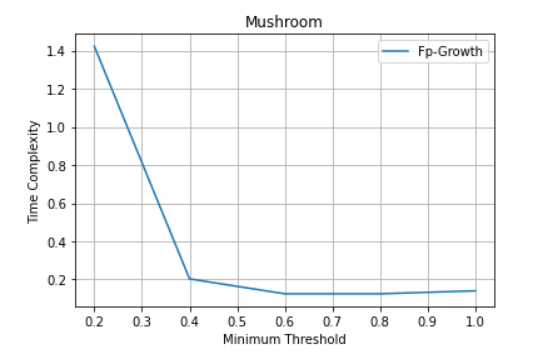


Figure 1: Execution time for various minimum threshold for mushroom data(all transactions) in FP-growth

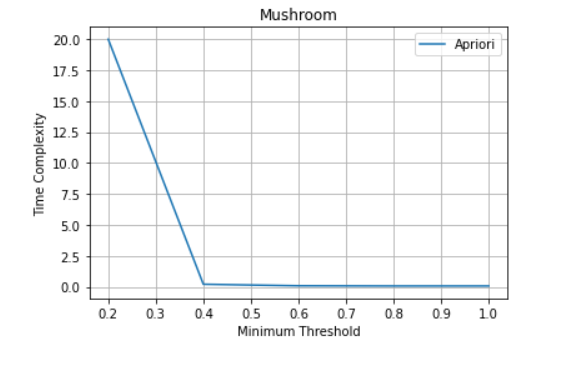


Figure 2: Execution time for various minimum threshold for mushroom data(all transactions) in Apriori

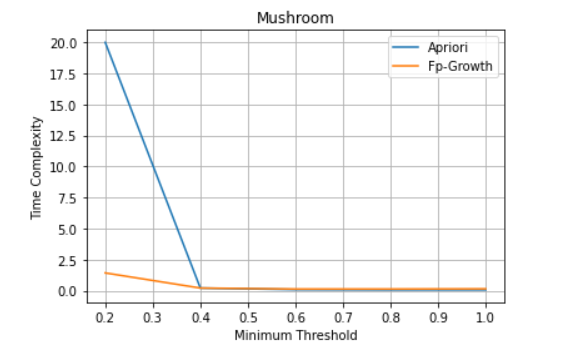


Figure 3: Execution time for various minimum threshold for mushroom data(all transactions) in FP-growth and Apriori



Figure 4: Execution time for various minimum threshold for chess data(all transactions) in Apriori

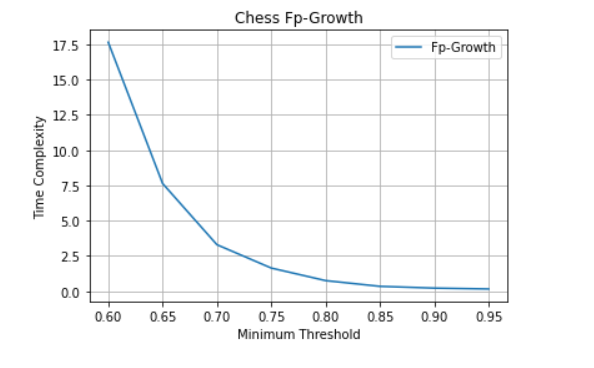


Figure 5: Execution time for various minimum threshold for chess data(all transactions) in FP-growth

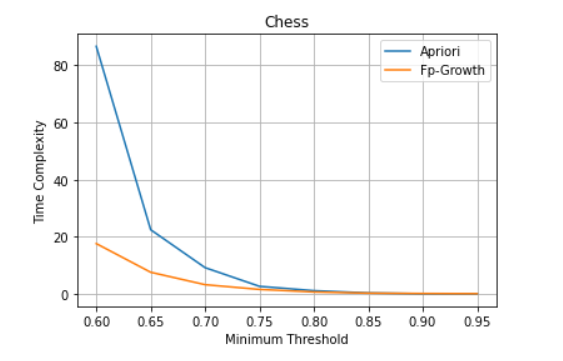


Figure 6: Execution time for various minimum threshold for chess data(all transactions) in FP-growth and Apriori

**Conclusion:**

After analysing the datasets we have comparison Apriori and fp-growth algorithms by their advantage disadvantage Memory and time complexity. Both two are complete algorithm and they can find frequent pattern efficiently. In apriori we have to sort the items in frequency descending order before using it to construct the tree. we can see from the graph, sorting with descending order is always faster. And the difference between the speed is more obvious with lower support. As a result more frequently occurring items will have better chances of sharing items. Thus The performance of fp-growth algorithm is higher than apriori as apriori generate candidate and scan every step.