COL761 Assignment 1 Q1

Kartik Arora (2021CS50124)　Piyush Chauhan(2021CS11010)

Ayushman Kumar Singh (2021CS51004)

February 10, 2025

§1. Introduction

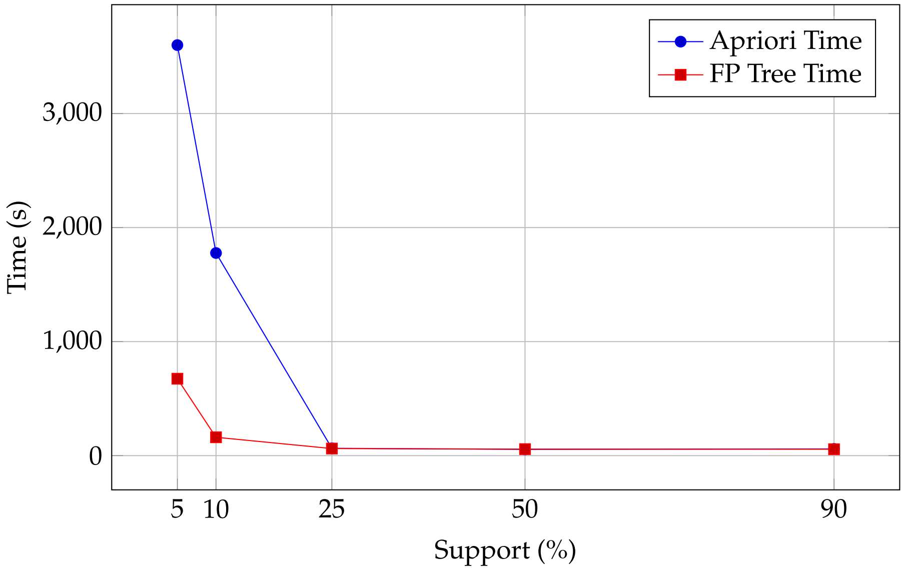
We conduct an empirical comparison of the Apriori and FP-tree algorithms for frequent itemset mining. Utilizing provided libraries, we analyze their efficiency on the given dataset.

§2. Results

Below are the run-times for the Apriori and FP tree algorithms.

Support	Apriori Time (s)	FP Growth Time (s)
5%	3600(TLE)	674.404
10%	1776.736	160.936
25%	64.285	62.006
50%	54.505	56.946
90%	58.726	56.115

Table 1: Comparison of Apriori and FP Tree Times



§3. Analysis

- At 5% support threshold (5%) , Apriori fails and gives TLE because it uses Breadth first search approach which generate many candidates.
- FP-tree also fails at 5% support and gives Memory Error because many items are frequent at such low support thus more conditional FP trees need to created. These conditional trees are almost as large as the original FP tree thus taking excessive memory. As FP tree works on depth first search approach, the recursion stack also contributes to memory consumption leading to MLE.
- In general, FP tree is faster than Apriori which can be seen at 10% support because it avoids candidate generationm and scans the dataset only twice as compared to multiple scans by Apriori.
- At high support (25%, 50% and 90%) , both algorithm have similar rumtime, with FP-Tree performing slightly better. This is because at high support candidate generated by Apriori are less.

§4. Conclusion

- FP-tree is significantly faster than Apriori, especially at lower support thresholds.
- Apriori is only efficient for high support thresholds or for smaller datasets.
- FP-tree may run into memory error at low thresholds because of excessive candidate generation.