Homework 2 Report

Microl Chen

February 23, 2023

1 Collaboration Statement

For this assignment, the following resources, help, or/and collaboration was utilized:
Overleaf - for the Latex Template
Stack Overflow - for optimization ideas/ pseudo codes
http://fimi.uantwerpen.be/src/ - borrowed optimization ideas from: A fast APRIORI implementation (FIMI03)
Debugged, diagnosed, and isolated errors with Programiz.com, Geeksforgeeks, Chatgpt, and Pycharm debugger. Lastly, "An improved Apriori algorithm for mining association rules" (DOI: https://aip.scitation.org/doi/abs/10 was also tremendously helpful.

2 Optimization Techniques & Notes

Since the data file we were provided contains 100,000 rows, the algorithm in the text book (or at least my implementation of it) was not efficient enough to run in 15 minute.

Upon looking at the algorithm's from http://fimi.uantwerpen.be/src/, several ways to optimize the algorithm was considered; upon those are the following ideas (notice that these ideas are very much inspired by Xiuli Yuan's ideas in the pseudo codes in the paper I cited):

1: Transaction Reduction - The general idea of transaction reduction is that transactions that did not contain subsets of any frequent should be removed.
2: Generation Reduction - One of the major problems with Apriori is that it over generates candidates, to achieve this, I examined techniques like item set hashing and more optimized pruning versions from other implementations.
3: Database Mapping - This technique allows us to avoid repeatedly scanning through the entire transaction list and is what ultimately allows for efficiency.

In the final implementation, a database dictionary was created to hold every transaction and every generated set that can be generated from existing sets that satisfies the minimum support. Unlike a traditional implementation where we generate candidate list with increasing sizes recursively and checks each list of candidates with the main transaction list, database mapping checks each generated candidate against the frequencies of the item sets in its previous sizes (which is conveniently stored in a dictionary) and not the entire list. More-ever, in addition to removing infrequent transactions, infrequent candidates were also removed.