CPSC4310/5310/7310
Spring 2023
Data Mining and Deep Learning

Course Project

Part II

3. Experiment and reports

3.1 Generate the databases as required in Section 2: D1K, D10K, D50K, and D100K.

3.2 A set of minimum supports (ms) are needed and could be 0.01, 0.05, 0.08, 0.10, 015. (Of course, sometimes you need to adjust them such that your implementation returns a set of frequent itemsets. But you should try your implementation with 5 different minimum supports.)

3.3 For the implementation of the Apriori algorithm, please run your implementation on each database using the set of minimum supports.

For each minimum support and for each database, put the frequent itemsets in the file corresponding to the minimum support and database, as required in Section 2. As another example, for a minimum support of 0.05 and the database D10K, the corresponding file for the frequent itemsets is D10K_Apriori_05.freq.

For each minimum support and for each database, you need to memorize and report the running time of the algorithm. For instance, at the end of the execution, you report the following message:

The frequent itemsets are stored in D10K_Apriori_05.freq, under ms = 0.05.
The time spent is 100s, to get the frequent itemsets.

3.4 For the implementation of Idea 1, similar to Section 3.3, run your implementation to get the frequent itemset files and the corresponding running times. In addition, you also report how many times you have had spent to scan the database. For instance, at the end of the execution, you report the following message:

The frequent itemsets are stored in D10K_Idea_05.freq, under ms = 0.05.
The time spent is 100s, to get the frequent itemsets.
The number of times scanning the database is 10, to get the frequent itemsets.

3.5 (For graduate students only) For the implementation of Idea 2, similar to Section 3.3, run your implementation to get the frequent itemset files and the corresponding running

times. In addition, you need to report the number of partitions you have used in the idea. You can use the same partition number for all the minimum supports and databases but you need to report the number as below. For instance, at the end of the execution, you report the following message:

The frequent itemsets are stored in D10K_Idea2_05.freq, under ms = 0.05.
The time spent is 100s, to get the frequent itemsets.
The number of the partitions used is 8.

4. Submission requirements

4.1 We do not care much of the arrangement of the files in this project. All the files are in the same folder. They include:

GenDatabase.cc or GenDatabase.py
apriori.cc or apriori.py
idea1.cc or idea1.py
idea2.cc or idea2.py
Necessary auxiliary files

Create a Makefile for compiling those source code files. The executables should be GenDatabase, apriori, idea1, and idea2. The marker can issue the following command:

%make

to compile and generate all the executables.

As examples, if the marker issues the following command:

%GenDatabase

the 4 databases as required in Section 2 are created in the same folder.

The marker can issue the following command:

%apriori D10K 0.05

to execute the implementation of the Apriori algorithm and the frequent itemsets are saved, as described in Section 3.

The marker can issue the following command:

%idea1 D10K 0.05

to execute the implementation of the Idea1 algorithm and the frequent itemsets are saved, as described in Section 3.

The marker can issue the following the command:

%idea2 D10K 0.05

to execute the implementation of the Idea2 algorithm and the frequent itemsets are saved, as described in Section 3.

4.2 Please write a project report, called ProjectReport.pdf, which should contain the following key points:

(a) Problem introduction, including what association mining is and what its applications are.
(b) The algorithms you implemented (High level idea, pseudo-code)
(c) Implementation issues (Data structures used, etc.)
(d) Experiment on a number of datasets with different parameters. (Create a table for each algorithm, report the execution times, and show some samples of frequent itemsets. You can use the above messages as a base and expand your writing.)
(e) Future work, challenges overcome and conclusion.

It is expected a report has around 4-5 pages.

4.3 You must submit your entire source code. It must have:

(a) All the related source code files, as described in Section 3.1, should be submitted in a USB. (The USB will be returned after marking.)
(b) The project report is put into the same folder. A hardcopy of the report is also needed for marking. The front page of your report needs to include the names of the group members. Attach the USB to the hardcopy report.

4.4 A demo will be arranged, depending on the time we have for the semester.