Data Mining Assignment 2

Submission Date & Time: 0900Hrs on 20/04/2017

Maximum Marks: 30

The goal of this assignment is to generate frequent item sets and Association rules using FP-tree.

Problem Description: Implement FP-growth algorithm using a programming language that you are familiar, such as C++ or Java.

Data: Download dataset file from https://archive.ics.uci.edu/ml/datasets/Pima+Indians+Diabetes

Apply FP-growth on this data to generate interesting frequent item sets and rules.

The input data now looks like this:

{1 3 4}
{1 2 5}
{1 2 6}

Run the code produce a tree in the given form:

(1) 1:3 (ref to null)
(1.1) 2:2 (ref to null)
(1.1.1) 5:1 (ref to null)
(1.1.2) 6:1 (ref to null)
(1.2) 3:1 (ref to null)
(1.2.1) 4:1 (ref to null)

Write a report to analyze the situations, such as data size, data distribution, minimal support threshold setting, and pattern density.

Support and confidence values are not fixed, experiment with different values of support and confidence.

After generating rules post your results in output file.

Programming languages: C, C++, JAVA
Report:
Report should contain following things:

1. ID and names of team members
2. Language used
3. What preprocessing was done to make it amenable for association rule mining

4. Compilation steps
5. Support and confidence value at which interesting rules are generated.
6. Number of rules generated

Submission Documents

1. Source code files, along with necessary files. The code should be read/write from files. You should not use stdin/stdout for input output purposes.
2. Report in pdf format
3. Output file as described above.

Remarks

1. All submission documents should be zipped together and submitted to CMS through one of the group member's account before deadline.
2. Although output files have to be given in submission, it should be reproducible when the code is executed again. Any discrepancies will result in loosing marks.
3. As said above, there should not be any IO from stdin/stdout. Your code should execute at one go after compilation. So please include necessary files in your submission folder. There will not be any attempt to debug your code by the evaluator.

Evaluation

Exact marks for evaluation will be disclosed later. But it will have following components:

1. Completion of code (copying code from friends or any resource will attract negative marks)
2. Successful compilation and execution
3. Association rules generated.
4. Report

Please contact following teaching assistants for any queries:

1. B. Rajitha (p2015409@hyderabad.bits-pilani.ac.in)