Assessment Descriptions

Assessment 3

Assessment name: Association rule/pattern mining for recommender system

Word count/length limit: report 12 pages
Weighting: 30%

CLOs addressed: 2, 3, 4, 5

Two milestones: 1) individual code due 9/04/2023 (end of week 10), 2) group report due 23/04/2023 (end of week 12)

Purpose: Assessment and practice of using association rule mining and recommender system methods.

The goal of assignment 3 is to apply pattern mining and recommendation system methods to solve a practical problem. **This is a group assignment**. You need to form a group of three.

Instructions for the assignment

This assignment consists of two main deliverables: runnable code and the report. The assignment should be done by a group of three students. To successfully accomplish this task, you need to be a good coder and demonstrate professional communication and writing skills.

Equal contribution and engagement of each group member is expected.

Milestone 3a: Individual code for job 1, 2 or 3 (5 points).

A. Suggested job description.

To start with, as a group you will decide on the work distribution and task allocations. As a suggestion, each person in the group may take on one of the following jobs:

1) Download the data, write the code in Python or R (Python is preferable) and run the code to mine frequent patterns from the training dataset. You can use any pattern mining algorithm discussed in the course, e.g. Apriori or FP-Growth. You can also use a method that is not discussed in the course, but a method description is required and why it is selected.
   Input: **data**, output: **patterns.**
2) Download the data, write the code in Python or R (Python preferred) to generate recommendations using collaborative filtering method from the training dataset. Your method should score recommendations for a user to select top recommendations. Select a metric to measure the performance. Write code to test the recommendations on the test set. This task will be used for two purposes: 1) produce recommendations from data, and 2) produce recommendations from patterns
   Input1: **data**, output: **recommendations.**
   Input2: **patterns**, output: **recommendations.**
3) Read provided references and conduct additional research to find at least one more credible academic source **on the use of frequent patterns to generate recommendations** or related research. Based on the research and your own ideas, write the code that integrates tasks 1) and 2) to produce results for the report.

B. Coding guidelines.

1. Individual code should be designed in such a way that each of these jobs can be done in parallel. Therefore, you need to agree on the code API, so it will be easier to integrate the code later to make the system.
2. Once individual part of the task in done, you will work together as a group to consolidate the code, generate the results, and write the report. You can consult group members about the job you are doing, but the code must be your own.
3. Task 3), as an individual code, should be tested without actual input from tasks 1) and 2). One way to do it is to simulate tasks 1) and 2) by providing dummy code of these two tasks, they are called stub methods (skeletons that simulate task 1) and 2).
4. Python libraries can be used for coding. If any external code is used, it should be adapted for the task, and cited in the code. For example, “this part is (based on) from [1]”. [1] is the reference number.
5. The required results are specified in the report section.

Milestone 3b: Formal Report (25 points) and final integrated code.

C. Report structure.

The report should be limited to 12 A4 pages including references, noting that including **necessary** contents has a priority over page limit.
The report should contain:

1. **Title page**: title of the project, names and ids of group members.
2. **Executive summary** (non-technical, <= 1 page). This section is for the company management that may not be familiar with technical details. It should include a brief problem description, benefits for the company and feasibility of scaling the solution. You can include test results, but need to explain what they mean for a layperson.
3. **Introduction (this starts the technical report)**: a brief explanation of the problem, the aim of the project.
4. **Exploratory analysis**: analysis of data that will give some insights how to use it, and potential solutions and potential problems that you may encounter. Diagram of the proposed system.
5. **Frequent pattern mining**:
    a. Brief description of frequent itemset mining method(s) and hyper-parameters used. If you choose a method that is not discussed in the course, method description is required and why it is selected, and a reference to a paper or a source.
    Selection of pattern as a guide: for association rules: confidence >=80%, for patterns: support >=5%. Choose your own thresholds to maximise benefit/profit for the company, justify your choice. Tip: make as parameters, as they affect the timing.
6. **Collaborative filtering** recommendation method
    a. Brief description of the method.
    b. Metrics used for evaluation on test dataset.
7. **Recommendation method from frequent patterns.**
    a. Brief description of the method.
    b. How this method is planned to work with patters/association rules as an input, and what will the output of this method.
8. **Discussion of results.**
    a. How the results were obtained, what metrics were used for evaluation. How the patterns and recommendations were ranked.

b. Five examples of frequent patterns with their confidence and support on both training and test sets.
c. 10 examples of recommendations from these patterns, two examples from each of the above patterns.
d. Table or chart of metrics with discussion, showing results of testing frequent patterns on the test set.
e. Table or chart of metrics with brief discussion, showing results of recommendations on training and test sets, **with and without frequent patterns used.**
f. Estimation of timing of the system if the dataset is scaled up to one million transactions.

9. **Conclusion and Recommendations**: which method do you recommend to use and why (recommendation from frequent patterns or directly from the dataset). Include scaling up consideration and benefits for the company. Include recommendations for future improvements.
10. **Reflection**: what is one main thing you have learned through this project and what would you do better next time.
11. **References (Harvard)**

D. Submission details

Milestone 3a: Individual code for job 1, 2 or 3 or according to your team work allocation (5 points).

- Submit one file, allowed extensions (.py, .ipynb or .r, .ipynb is best as it allows to make nice comments)
- Do not include the dataset in the submission.
- **Every submitted file name must be in the form "<your_id>_<your name>_<group_number>_assign3".**

Milestone 3b: Group report and final integrated code.

- Do not include the dataset in the submission.
- Please submit two files (do not zip):
  - One code file **"<group_number>_assign3",** allowed extensions (.py, .ipynb or .r)
  - report as pdf file **"<group_number>_assign3.pdf".**
- The final code can be modified to produce extra results for the report, but no major rewriting of the code should be done.

Late submission rules:

Milestone 1: late submissions will receive 0 marks. The code will be marked from this submission, not the one with the report.

Milestone 2: If you hand in your work late, your mark will be capped, based on the number of late days. A part of the late day is counted as full day.

– 1 day late – mark capped at 75%

– 2 days late – mark capped at 50%

– 3 days late – mark capped at 25%

– more than 3 days late – no marks available

Submitting the final code is essential to mark the report.

E. Suggested reference papers:

Lee, C.H., Kim, Y.H. and Rhee, P.K., 2001. Web personalization expert with combining collaborative filtering and association rule mining technique. Expert Systems with Applications, 21(3), pp.131-137. https://www.sciencedirect.com/science/article/pii/S0957417401000343/pdfft?casa_token=68QbbDO8URAAAAAA:cmETEPCp3pOkzxZC_vF_dKDgU41RwIyj1ZpQgGwkvNjIMVyCCZLGNYWUVvPr-HZCNwSDrAhGGIA&md5=c9cc1e9206467a1b3df9d70ad12f93b7&pid=1-s2.0-S0957417401000343-main.pdf

Parvatikar, S. and Joshi, B., 2015, December. Online book recommendation system by using collaborative filtering and association mining. In 2015 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC) (pp. 1-4). IEEE. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7435717

Academic Integrity Declaration

By submitting this assignment, I declare that this assessment item is my own work, except where acknowledged, and has not been submitted for academic credit elsewhere. I acknowledge that the assessor of this item may, for the purpose of assessing this item, reproduce this assessment item and provide a copy to another member of the University; and/or communicate a copy of this assessment item to a plagiarism checking service (which may then retain a copy of the assessment item on its database for the purpose of future plagiarism checking). I certify that I have read and understood the University Rules in respect of Student Academic Misconduct.