

Contemporary Computer Science: Recommender Systems

Coursework: Summative Assignment

Lecturer

Suncica Hadzidedic

suncica.hadzidedic@durham.ac.uk

E231

Level: 3 and 4

Date of distribution: 31 January 2020

Date of submission: **20 March 2020 (5 p.m.)**

Workload: 8-9 days, 3-4 h/day

Grading method: report, application

Total marks: 100 (L3 students), 150 (L4 students)

Submission via DUO to include, in a compressed folder (.zip/.rar):

- Report (PDF file)
- Source code, resource files (including datasets), readme .txt file

Building a context-aware recommender system

As part of the Recommender Systems module, you are required to submit coursework which is part of the summative assessment and worth 50% of your grade. You are to develop a **context-aware** music or travel recommender system application. Note that the focus of this assignment is the implementation of a recommender system technique.

L3 students will be marked on Criteria A in Table 1, with a total mark of 100. Level 4 students will be marked both on Criteria A and B in Table 1, totalling 150 marks.

Coursework requirements

Components 1-3 are requirements for both L3 and L4 students.

1. System development

- Programming language: Python

- Note: You can reuse and adjust any material you have developed for the Web Technologies module in the previous year/term.
- Dataset: you can use an existing dataset (e.g. #nowplaying-RS¹, MusicMicro², other CARS datasets³), synthetically generate a dataset (e.g. using DataGenCARS), or generate one from real usage data.
- Context: select a single context (e.g., mood) from the contextual information collected in the dataset.
- Methods/techniques:
 - You are to research, select, justify (in the report), and apply to your system techniques for the following:
 - i. Feature extraction and selection -> user data considered, item data considered, and context selected;
 - ii. 2D/MD recommender technique (user profiling/modelling);
 - iii. Contextual recommendation generation: contextual information filtering/modelling technique (e.g., pre- or post- filtering, neural collaborative filtering, etc.).
- User interface: Develop a simple UI for your application – this can be a Python script for input and output. You are to showcase:
 - *Input*: How does the system recognise the active user? Which user data is gathered – explicit/implicit?
 - *Output*: How are recommendations and/or predictions presented to the user? Consider target user needs, in terms of e.g.: number of recommendations presented; style of presentation; user context – environment, device used; user characteristics – age, disabilities, etc.
- Make sure to reference any external sources you have used for datasets, code or algorithm logic.

2. Evaluation

- Evaluate the performance of your RS by carrying out an offline experiment.
- Apply all of the following measures:
 - accuracy of ratings predictions (MAE), accuracy of usage predictions (precision, recall)
- Present (in the report) system performance results and compare against benchmarks (other related work in literature)
 - Select and explain which of the measures reflect the needs of the domain of application.

3. Report

Write a report on the implementation of your CARS, addressing all of the following:

- Formatting: 2-3 pages, use IEEE conference paper formatting⁴

¹ <https://zenodo.org/record/3248543#.Xhvj385zyM8>

² <http://www.cp.jku.at/datasets/musicmicro/index.html>

³ https://github.com/irecsys/CARSKit/tree/master/context-aware_data_sets

- Introduction
 - Domain
 - Related work review
 - Purpose / Aim
- Methods
 - Data type, source
 - Feature extraction and selection methods
 - User profiling and prediction methods
 - Evaluation methods
- Implementation
 - Recommendation algorithm
 - Output (recommendations/predictions) presentation
- Evaluation results
- Conclusion
 - Limitations
 - Further developments
- References

4. Additional requirements (**only L4**)

L4 students are required to perform additional work worth 50 points, as follows:

- *System development*
 - Apply one additional context
 - Implement recommendations based on **two** context types (e.g., mood and location).
 - Apply one additional contextual information filtering/modelling technique to your system (**two** techniques in total)
 - User interface
 - Consider recommendations' transparency. Within the output presentation, include explanations for why that specific set of recommendations is presented to the user.
- *Evaluation*
 - Additionally apply the following measures and present their results in the report:
 - novelty, accuracy of ratings predictions (RMSE), accuracy of usage predictions (ROC), coverage
 - Note: You are to apply the 7 evaluation metrics to only one of the system versions, i.e., one of the contextual information modelling techniques.
 - Compare the performance of the two contextual information modelling techniques you have implemented:
 - Select **one** performance evaluation metric to compare on
 - Present the evaluation results in the report

⁴ <https://www.ieee.org/conferences/publishing/templates.html>

Marking Specification

Table 1 specifies the distribution of marks (100, i.e. 150 for L4). Make sure you covered all the elements you will be marked on as listed in the table.

Table 1. Marking Specification

<i>Criteria A (L3 and L4 students)</i>	Mark (/100)
CARS Design and Implementation	45
Methods/techniques applied (features, contextual information, user profile, prediction)	35
Interface (input, output)	10
CARS Evaluation	35
Comprehensiveness of system evaluation: 3 metrics	15
Presentation/comparison of metrics and relevance to domain	20
Report	20
<i>Criteria B (L4 students: Additional components)</i>	Mark (/50)
CARS Design and Implementation	26
Additional context and modelling technique	19
Output: Explanation for recommendations	7
CARS Evaluation	24
Comparison of modelling techniques	8
Application of additional evaluation metrics	16