Durham University

Computer Science

Summative Assignment

Module code and title	COMP3607 Recommender Systems
Academic year	2023/24
Submodule title	N/A
Coursework title	Recommender Systems Assignment
Coursework credits	10 credits
Lecturer	Suncica Hadzidedic
Deadline*	Thursday, January 25, 2024 14:00
Hand in method	Ultra, Panopto video

Additional coursework files	• video • source code – Jupyter Notebook
Required submission items and formats	Submit via **Ultra**: • Source code • Panopto video

*** This is the deadline for all submissions except where an approved extension is in place. Late submissions received within 5 working days of the deadline will be capped at 40%. Late submissions received later than 5 days after the deadline will received a mark of 0.**

COMP3607 Coursework:

Designing and Developing Personalised Recommender Systems

Overview

Lecturer/Marker

Suncica Hadzidedic
suncica.hadzidedic@durham.ac.uk
Room MCS1034

Hand-out to students: 27 October 2023

Type: summative assessment
Level: 3
Components marked: video, code
Total marks: 100
Weight of module mark: 100%
Expected workload (formative + summative): 40 h

Submission instructions

Submission deadline	**25 January 2024; 14:00**
Format	• **Ultra - Panopto:** video[1]. • **Ultra** – compressed (.zip or .rar) files to include: source code in Jupyter Notebook with printed outputs
Plagiarism, collusion	Your work must be done by yourself and comply with the university rules about plagiarism and collusion: https://www.dur.ac.uk/learningandteaching.handbook/6/2/4/

[1] Instructions for creating videos provided on our Ultra page, within the submission point.

I. Requirements

A. You are required to design, develop and evaluate:
   i. **RS1** (40% of the mark): At the **basic** level, one personalised **conventional** recommender system (e.g., content-base filtering, hybrid RS, collaborative filtering).
   ii. **RS2** (40% of the mark): To demonstrate **additional effort**, extend the conventional RS with more advanced or state-of-the-art methods, e.g., deep learning, knowledge graphs, reinforcement learning, LLMs.

B. You can choose any **domain** and **dataset** for your RS. You have to use the same dataset/domain for both RS implementations. A list of some publicly available datasets is provided at the end of this document (Section IV). You are free to use other datasets or synthetically generated data.

C. You will present your work in a 10-minute video, that will include:
   i. a video and audio recording
   ii. presentation (ppt), system and code demo.

D. Your coursework should meet the following requirements:

1. **Programming language and testing environment**
   - You must implement your RS in Python. You have to be able to demonstrate a functional recommender system that takes user input, processes data, and presents output in a user interface. You do not need to submit these files.
   - For code review purposes, you will submit your code in a **Jupyter Notebook**, with the printed outputs at different stages (i.e., data preparation, evaluation metric results, recommendations, etc.)
   - Your RS solution will be tested on: laptop (2.8 GHz, 8 GB RAM); Windows 10 OS; Anaconda 3.
   - Make sure to reference any external sources you have used for the code, data, algorithm logic, etc.

2. **User interface**
   - This should be a **command line** interface for both RS versions.
     - Note: Do not develop graphical, web-based interfaces. These will not be marked.
   - Take into account the following:
     - *Input UI*: How does the system recognise the active user?
     - *Output UI*: Number of recommendations presented; style of presentation.

3. **RS1 – basic level: Personalised recommender system with conventional methods**

   i. **RS technique:** You are free to choose any conventional RS technique for your personalised RS (e.g., CF with matrix factorization, CBF with TFIDF, 2D CARS, hybrid of two conventional techniques, knowledge-based RS).

- You have to provide a **justification** for the suitability of the selected RS technique for the RS purpose, domain of application and available data.

ii. **Dataset:** Select any dataset suitable for the RS you intend to develop. However, you have to use the same dataset for both RS1 and RS2. For RS1:
  - Randomly sample 100,000 (100K) cases from the original dataset.
  - Use this 100K dataset for the RS1 implementation.
  - Describe the dataset and explain the data preparation methods in sufficient detail in the video.

iii. **Methods**: You should research and select the most appropriate/suitable methods for your system, including those for: user profile learning, rating prediction, evaluation metrics, etc.
In the video presentation, **describe** and **justify** the methods.
  - Justifications should address the selected methods' relevance to and alignment with the purpose, application domain and data used for the implemented recommender systems.
  - Cite supporting literature.

iv. **Evaluation**: Evaluate the performance of your conventional RS by carrying out an **offline experiment,** as follows:
  - Choose **<u>one</u> evaluation metric** from the two categories listed below:
    1. Accuracy of **rating** predictions
    2. Accuracy of **ranking**.
  - **Justify** in the video presentation the selected metric, i.e., why the metric is appropriate for the purpose of the RS and domain of application, with supporting references.
  - Apply the metric you have chosen to evaluate your conventional RS.
  - Present and interpret the results for the metric:
    - Clearly present (e.g., in tables or graphs) and interpret the results in the video.
    - Demonstrate in the code that you have applied the evaluation metric.

4. RS2 – additional effort: Personalised recommender system with advanced methods

i. **Methods:** You are required to do **research** and expand your baseline, i.e., conventional RS technique, with advanced, state of the art, methods. The **state-of-the-art** methods should be applied to **<u>one</u>** of the following:
  - user preference/rating prediction; use of multimodal data; feature extraction; explanations of recommendations.

In the video presentation, **describe** and **justify** the methods.
  - Justifications should address the selected methods' relevance to and alignment with the purpose, application domain and data used for the implemented recommender systems.
  - If possible, illustrate the recommendation techniques/process.
  - Cite supporting literature.

ii. **Datasets:** For RS2, you will:
- Use the 100K dataset that you used for the RS1 implementation.
- Explain the data preparation methods in sufficient detail in the video.
- Train and test RS2 on the 100K dataset.

iii. **Evaluation**: Carry out an **offline experiment** and evaluate the performance of your advanced RS.
- Choose **one** additional metrics from the following:
  - novelty, explainability, fairness.
- **Justify** in the video presentation why the metric is appropriate for the purpose of the RS and the domain of application.
- Apply the metric to your two RS versions:
  - Conventional RS (baseline)
  - Advanced RS
- **Compare** the performance of RS1 and RS2 on:
  - **Both** evaluation metrics you have chosen (the first metric from RS1 requirements, and the additional metric from the RS2 requirements).
- Present and interpret the results for both metrics:
  - Clearly present (e.g., in tables or graphs) and interpret the results in the video.
  - Demonstrate in the code that you have applied the evaluation metrics.
  - Draw conclusions about the advanced RS performance.

5. Video

The video should showcase your overall work.

i. **Formatting**:
- It has to be up to **10 minutes** long.
- It has to include video and audio recording. Videos without an audio recording will have a penalty.
- Use presentation slides (preferably Power Point) to report about your work.
- Demonstrate that your RS versions are functional by running your RS and showing the code.
- You are free to distribute the 10 minutes between the presentation and the system demo as you see fit.

ii. The **presentation slides** have to cover the following content:
- Introduction
  - Domain of application
  - Purpose/Aim
- Methods and their justification – for both RS1 and RS2
  - Data description
  - Data preparation
  - Recommendation techniques/algorithms
- Evaluation – for both RS1 and RS2
  - Evaluation metrics (in equations) and justification

- Evaluation results (in tables or graphs)

- RS1 and RS2 comparisons and conclusions (about RS performance)
- References in IEEE style

iii. **RS demo -** You will demonstrate that your system (both the conventional and advanced) is functional through an example of **one** user, whereby you will showcase:

- Input interface: how the user inputs data to the system.
- Output interface: how recommendations are presented to the user and any other interaction a user is allowed at this stage.
- Back-end: by **going through the code** to explain the main:
  - techniques/algorithms used
  - RS evaluation metrics and performance results.

Note1: Marks will be **reduced** for videos longer than 10 minutes. For every 5 seconds longer, 10% of the marks for the video will be reduced.

Note2: For videos without an audio recording - the *Presentation* mark will be reduced by 50%.

II. Marking Criteria

The distribution of coursework marks is presented in the table below.

<table>
<tr><td>RS1. Conventional RS – design and development</td><td>30</td></tr>
<tr><td colspan="2">The following methods were explained and justified:
• Appropriate dataset for domain used; 100K size
• Data preparation
• Appropriate conventional RS technique
• Presentation of recommendations (style, number)
RS implementation:
• All the reported methods were implemented
• RS was implemented in Python; code submitted in Jupyter Notebook
• RS is functional, without errors
• Input interface – active user is recognised
• Output interface – personalised recommendations are presented</td></tr>
<tr><td>RS1. Conventional RS – evaluation</td><td>10</td></tr>
<tr><td colspan="2">• One ranking or rating accuracy evaluation metric was selected
• Clear and supported justification for the suitability of the metric was provided
• The accuracy metric was applied to the conventional RS, tested on 100K dataset
• Evidence of applied evaluation metric is provided in Jupyter Notebook
• Results were presented and interpreted clearly</td></tr>
<tr><td>RS2. Advanced RS – design and development</td><td>25</td></tr>
<tr><td colspan="2">Conventional RS was expanded with state-of-the-art methods.

Appropriate state-of-the-art methods were selected, explained and justified for one of the following:
• preference/rating prediction, use of multimodal data, feature extraction, recommendations' explanations</td></tr>
</table>

<table>
<tr><td colspan="2">Advanced RS implementation:
All the selected methods were implemented
RS is functional, without errors
RS was trained and tested on 100K dataset
Output interface – personalised recommendations are presented</td></tr>
<tr><td>RS2. Advanced RS – evaluation</td><td>15</td></tr>
<tr><td colspan="2">One metric was selected from novelty, explainability, fairness
Clear and supported justification for the suitability of the metric was provided
Advanced RS was evaluated on: 100K dataset, two metrics
Conventional and advanced RS were compared on: 100K dataset, two metrics
Evidence of applied evaluation metrics is provided in Jupyter Notebook
Results were presented and interpreted clearly</td></tr>
<tr><td>Presentation (video)</td><td>10</td></tr>
<tr><td colspan="2">Format: 10 minutes, includes presentation slides and system demo, includes audio and video recording
Required content covered in the presentation slides and RS demo
Presentation style – clarity, confidence, creativity, attractive design, pace.
Penalty: marks reduced for videos longer than 10 minutes
Penalty for videos without audio recording</td></tr>
<tr><td>Participation</td><td>10</td></tr>
<tr><td colspan="2">10% of your coursework mark will be derived from your on-going participation and engagement in the module’s activities and tasks, including: formative assignment, peer review, in-class tasks, group discussions. Marking is based on submission (1 mark) or non-submission (0 marks).
Formative assignment - 40%
Peer review - 20%
Weekly tasks - 40%</td></tr>
<tr><td colspan="2" align="right">TOTAL /100</td></tr>
</table>

III. Learning Outcomes

Subject-specific knowledge demonstrated via:

- an understanding of the different types of recommender systems, their purpose and domains of application
- an understanding of recommender system users: usage behaviour, demographics, preferences, contextual information
- an in-depth knowledge of recommender system algorithms.
- an understanding of recommender system evaluation methods.

Subject-specific and key skills demonstrated via:

- an ability to undertake self-study and independent research
- an ability to critically analyse and evaluate state of the art practices
- an ability to apply RS methods and techniques
- an ability to implement a recommender system for a specific domain
- an ability to evaluate RS performance.

IV. Datasets

Some publicly available datasets for recommender systems are listed here:

Dataset	Link	Description
Yelp	https://www.yelp.com/dataset	user reviews of different businesses and services in a specific location
ReDial	https://redialdata.github.io/website/	Dialogues of users recommending movies to each other
LDOS-CoMoDa	https://www.lucami.org/en/research/ldos-comoda-dataset/	context-aware movie data
Million Song	http://millionsongdataset.com/	music and context data
Last.fm	http://millionsongdataset.com/lastfm/	song tag and song similarity
GroupLens	https://grouplens.org/datasets/	movies, books, personality-aware data
Inspired	https://github.com/sweetpeach/Inspired	1,001 human-human dialogs for movie recommendation
UC San Diego	https://cseweb.ucsd.edu/~jmcauley/datasets.html	RS datasets for Amazon reviews, Goodreads, clothing data, etc.
KuaiRand	https://kuairand.com/	Videos - unbiased sequential recommendation
KuaiRec	https://kuairec.com/	Kuaishou videos - fully observed user-item interaction matrix
Criteo Click Logs	https://ailab.criteo.com/download-criteo-1tb-click-logs-dataset/	Ads - feature values and click feedback
Epinions	https://www.cse.msu.edu/~tangjili/datasetcode/truststudy.htm	User profile, ratings and trust relations
Ciao	https://paperswithcode.com/dataset/ciao	Trust in RS: rating information of users given to items, and also contain item category information
Douban	https://paperswithcode.com/dataset/douban	social network: user review and recommendation services for movies, books, and music
Taobao	https://www.comp.hkbu.edu.hk/~lichen/download/TaoBao_Serendipity_Dataset.html	e-commerce, user perception of recommendation, curiosity and personality
WeChat	https://github.com/yaqingwang/WeFEND-AAAI20	Users and labeled news
Synthetic data generation	e.g. DataGenCARS - http://webdiis.unizar.es/~maria/?page_id=70	
Social and health data	UK Data Service	
StudentLife	https://studentlife.cs.dartmouth.edu/dataset.html	sensor data, EMA data, survey responses and educational data
Snapshot Study	https://www.media.mit.edu/projects/snapshot-study/overview/	Students - physiological, behavioral, environmental, and social data using mobile phones, wearable sensors, surveys, and lab studies

V. FAQs

Question: To what degree should we implement the recommenders “from scratch”? I am assuming using sklearn models for classification is ok, but what about a library that implements factorization machines?

Response: You can do most of the implementation by using existing packages, modules, libraries.

For the user interface should the user be able to provide ratings and change the dataset?

R: User profile updating is not a must, however, that does increase the complexity of your RS, considered an advanced option.

Are we allowed to bring in extra data to a dataset (for example the MovieLens dataset does not have tag data but does contain links to IMBD pages where we can extra descriptions) ?

R: Yes, you can use additional data - merge different datasets, as well as create synthetic datasets.

I am really interested in the DataGenCARS http://webdiis.unizar.es/~maria/?page_id=70 software listed in the assignment brief however I have been unable to get hold of the author. Do you know of any other ways to access this?

R: The only way I know about it is by contacting the authors. Please send them a reminder email, if they have not responded they might just be busy. However, in the meantime, there are other tools/options of synthetic data generation for RSs - you can search for those, and contact their authors. I advise you not to lose too much of your time on this approach to data collection/procurement.

I have been working with a large dataset (over 1.5GB). This takes a great deal of time to process initially and amend records, however, if I 'trim' the dataset and make it a more reasonable size, then I will lose accuracy with my models. In any case, what would you recommend for the coursework?

R: Your coursework solution and results will not be assessed on how accurate your system is. The essential part is that you follow the requirements, develop the RS , evaluate and compare the performance results. How well-performing they are is not relevant, as long as you can discuss those results. Therefore you can reduce the size of the dataset, if that will help you.

Are we expected to be able to add new users?

R: No, but you can. However, consider whether this contributes sufficiently to your system, or if it is better to put your efforts into improving/optimizing the system in some other area.

For the video part of the coursework, is it okay to edit and skip certain sections i.e., querying and updating my datafiles as well as computing matrices takes a while? If I am able to skip the computation, I will have more time to explain the systems' functionality.

R: Yes, of course. The time it takes to train/compute/update your system does not need to be included in the video.