CP421 Data Mining - Project
Due Date: Dec 9th, 2020 at 11:00 PM

To bring together and apply the various topics covered in this course, in this project you will work on a recommendation system and make movie recommendations to users. You are free to explore different strategies to improve the performance of your system.

1 Deliverables

There are several deliverables for your project that will be graded team based to make up your final project score.

- **Project Presentation** Each group will explain their project in a 15-minute presentation to the class. Presentations should clearly convey the project ideas, methods, and results, including the question(s) being addressed, the motivation of the analyses being employed, and relevant evaluations, contributions, and discussion questions.

- **Project Paper** The projects will be concluded with a project paper. Your paper should summarize your steps in developing your solution, including how you processed the data, the method you used, and the insights obtained. The paper should be submitted in Word or PDF format. Optionally, you may submit a draft paper (in person or via email) by the listed deadline to receive feedback from the instructors prior to the final paper deadline.

2 Grading

50% of the project grade will be based on your project paper. 50% of the grade will be based on your project presentation. The project presentation will be graded as follows:

Introduction:	15%	Provide context. What questions are being addressed?
Solution/Method:	30%	What did you do? Why did you choose this method? What tools and techniques did you use?
Data and Experiments:	10%	What data did you use? Are your experimental methods reliable?
Evaluation and Results:	30%	What evaluation did you do? Do your conclusions match your results?
Presentation Quality:	15%	Clarity of speaking (5%), organization (5%), and visuals (5%).

The project paper will be graded as follows:

Introduction:	15%	Provide context and motivation.
Related Work:	10%	How do baseline methods differ from your method?
Solution/Method:	25%	What did you do? What tools and techniques did you use?
Data and Experiments:	10%	What preprocessing was done to the data?
Evaluation and Results:	25%	How did you evaluate your method? What is your conclusion?
Writing Quality:	15%	Clarity of writing, organization, and grammar.

3 About submission

To submit your project deliverables, create a folder named *login_pj* and place your files in this folder. The folder should include your project paper (Word or PDF format), presentation materials, and all code and output used to generate results. Compress the folder (please use .zip compression) and submit it to MyLearningSpace. Please submit one file per team and include the names of all your team members in the report.

4 Data

4.1 General Description

You will be using the *ml-latest-small* dataset that describes 5-star rating and free-text tagging activity from MovieLens [1], a movie recommendation service. It contains 100836 ratings and 3683 tag applications across 9742 movies. These data were created by 610 users between March 29, 1996 and September 24, 2018. This dataset was generated on September 26, 2018, and can be downloaded at

```
http://files.grouplens.org/datasets/movielens/ml-latest-small.zip
```

Users were selected at random for inclusion. All selected users had rated at least 20 movies. No demographic information is included. Each user is represented by an id, and no other information is provided. The data are contained in the files links.csv, movies.csv, ratings.csv and tags.csv [2]. **While developing your method, you are free to use some or all of the data files.**

4.2 Formatting and Encoding

The dataset files are written as comma-separated values files with a single header row. Columns that contain commas (,) are escaped using double-quotes ("). These files are encoded as UTF-8.

User Ids MovieLens users were selected at random for inclusion. Their ids have been anonymized. User ids are consistent between ratings.csv and tags.csv (i.e., the same id refers to the same user across the two files).

Movie Ids Only movies with at least one rating or tag are included in the dataset. These movie ids are consistent with those used on the MovieLens web site (e.g., id 1 corresponds to the URL https://movielens.org/movies/1). Movie ids are consistent between ratings.csv, tags.csv, movies.csv, and links.csv (i.e., the same id refers to the same movie across these four data files).

Ratings Data File Structure (ratings.csv) All ratings are contained in the file ratings.csv. Each line of this file after the header row represents one rating of one movie by one user, and has the following format:

```
userId,movieId,rating,timestamp
```

The lines within this file are ordered first by userId, then, within user, by movieId. Ratings are made on a 5-star scale, with half-star increments (0.5 stars - 5.0 stars). Timestamps represent seconds since midnight Coordinated Universal Time (UTC) of January 1, 1970.

Tags Data File Structure (tags.csv) All tags are contained in the file tags.csv. Each line of this file after the header row represents one tag applied to one movie by one user, and has the following format:

```
userId,movieId,tag,timestamp
```

The lines within this file are ordered first by userId, then, within user, by movieId. Tags are user-generated metadata about movies. Each tag is typically a single word or short phrase. The meaning, value, and purpose of a particular tag is determined by each user. Timestamps represent seconds since midnight Coordinated Universal Time (UTC) of January 1, 1970.

Movies Data File Structure (movies.csv) Movie information is contained in the file movies.csv. Each line of this file after the header row represents one movie, and has the following format:

```
movieId,title,genres
```

Movie titles are entered manually or imported from https://www.themoviedb.org/, and include the year of release in parentheses. Errors and inconsistencies may exist in these titles. Genres are a pipe-separated list, and are selected from the following:

```
Action
Adventure
Animation
Children's
Comedy
Crime
Documentary
Drama
Fantasy
Film-Noir
Horror
```

[1] https://movielens.org/

[2] http://files.grouplens.org/datasets/movielens/ml-latest-small-README.html

```
Musical
Mystery
Romance
Sci-Fi
Thriller
War
Western
(no genres listed)
```

Links Data File Structure (links.csv) Identifiers that can be used to link to other sources of movie data are contained in the file links.csv. Each line of this file after the header row represents one movie, and has the following format:

```
movieId,imdbId,tmdbId
```

movieId is an identifier for movies used by MovieLens. E.g., the movie Toy Story has the link

`https://movielens.org/movies/1.`

imdbId is an identifier for movies used by IMDB. E.g., the movie Toy Story has the link

`http://www.imdb.com/title/tt0114709/.`

tmdbId is an identifier for movies used by TheMovieDB. E.g., the movie Toy Story has the link

`https://www.themoviedb.org/movie/862.`

Use of the resources listed above is subject to the terms of each provider.

5 Proposal

You are expected to develop your own version of recommendation system. Some of the options can be but not limited to

- Take the matrix factorization method, and implement SGD to derive the parameters;
- Apply natural language processing methods on tags, and infer user preference in the history;
- Exploit time stamps embedded in data files, and capture the temporal relations between ratings;
- Analyze movie genres, and adjust users' weighting in collaborative filtering with different users expertise.

6 Experiment

To demonstrate the effectiveness of your proposal, you are supposed to compare your method with multiple baselines. You can start with some simple methods listed below.

- Use the global mean to do the prediction;
- Use the user means as the prediction;
- Use the item means as the prediction;
- Use the classic method discussed in class.

Training and testing data To perform an evaluation, you need to first split the data into training and testing datasets. Using the training data, you can develop a recommender system with your proposed method. For the testing data, you can randomly select some cells from the user-item matrix, e.g., 20%, and assume that they are unknown. Then by feeding the testing data into the recommender system, it can estimate ratings for the missing cells. Finally, you can compare the system outputs against the real values.
Performance Evaluation You will use the root mean-square error (RMSE) to measure the performance.

7 Sample code

We provide basic sample code to load, preprocess, and analyze the data. For more information, please refer to

`https://github.com/wlucp421/project/blob/master/samplecode.ipynb`