DNN-MF Recommender System

This repository contains the code for a DNN-MF recommender system. The system is designed to recommend movies to users based on their previous ratings. The recommender system is built using Python, and leverages the power of deep learning with the Keras library.

Table of Contents

1.0-initial data exploration.ipynb

This notebook is used to explore and visualize the initial dataset. It reads in the raw data and preprocesses it into a format suitable for the recommender system. It also generates several plots to provide insights into the data.

2.0-data-preprocessing.ipynb

In this notebook, we preprocess the data. This includes standardizing the user's age, scaling the ratings, and splitting the data into training and testing sets. The preprocessed data is then saved for later use.

3.0-DNN-MF-model.ipynb

This notebook contains the implementation of the DNN-MF model. The model is trained on different subsets of the data separately and in turn, while maintaining weights. The model is then evaluated on a test set.

4.0-Test-function.ipynb

This notebook is used to test the recommender system. It loads the trained models and uses them to make predictions on a test set. The predictions are then compared to the actual ratings to evaluate the performance of the recommender system.

evaluate.py

This script is used to evaluate the performance of the recommender system. It calculates the root mean square error (RMSE) and precision at 10 for the MF and DNN_INFO models.

How to run the code

1. Clone the repository to your local machine.
2. Install the required Python libraries.

3. Run the Jupyter notebooks in the following order:
   - 1.0-initial data exploration.ipynb
   - 2.0-data-preprocessing.ipynb
   - 3.0-DNN-MF-model.ipynb
   - 4.0-Test-function.ipynb
4. Run the evaluate.py script.

If You don’t want to retrain the model, don’t run the 3.0-DNN-MF-model.ipynb notebook.
If you retrain the model, rewrite the downloading the model in the 4.0-Test-function.ipynb and evaluate.py(write the last saved checkpoints)