Assignment 4: CS446 Final Project

Loïc Dallaire William Forget

April 20, 2025

Abstract

We propose a hybrid recommender system that reframes movie rating prediction as a five-class classification task and integrates collaborative and content-based filtering. Our multi-layer perceptron (MLP) classifier, whose hyperparameters are optimized via a genetic algorithm (GA), achieves 88% accuracy on the MovieLens test set, outperforming a baseline MLP regressor. We also explore metaheuristic hyperparameter search, albeit without significant gains.

Introduction

The MovieLens dataset, compiled by the University of Minnesota's GroupLens Research Project (led by John Riedl and Joseph Konstan), comprises 100,000 ratings (1–5 stars) from 943 users on 1,682 movies. Each participant rated at least 20 titles and provided basic demographics. Collected between September 19, 1997 and April 22, 1998 via the MovieLens website and cleaned to remove incomplete profiles, it is freely available for research. To address the tendency of regression models to predict mean ratings under skewed distributions, we treat rating prediction as a multiclass classification problem and develop a hybrid recommender system combining collaborative and content-based methods, implemented through an MLP classifier tuned by a GA.

Objective

The primary objective of this project is to build a hybrid recommender system that leverages:

1. A multi-layer perceptron (MLP) for predicting user ratings as discrete classes (1–5 stars).
2. A genetic algorithm (GA) for optimizing the hyperparameters of the MLP model.

Dataset and Preprocessing

- Load raw data (`u.data` for ratings, `u.item` for movies)
- Clean and convert types (e.g. cast `movie_id` to integer)
- Merge ratings with movie metadata on `movie_id`
- Filter out duplicate records
- Select only the columns of interest (user, movie, rating, genres)
- Transform the `genre` flags into a one-hot matrix
- Optionally split into training and testing subsets
- Prepare features for content-based and collaborative filtering models

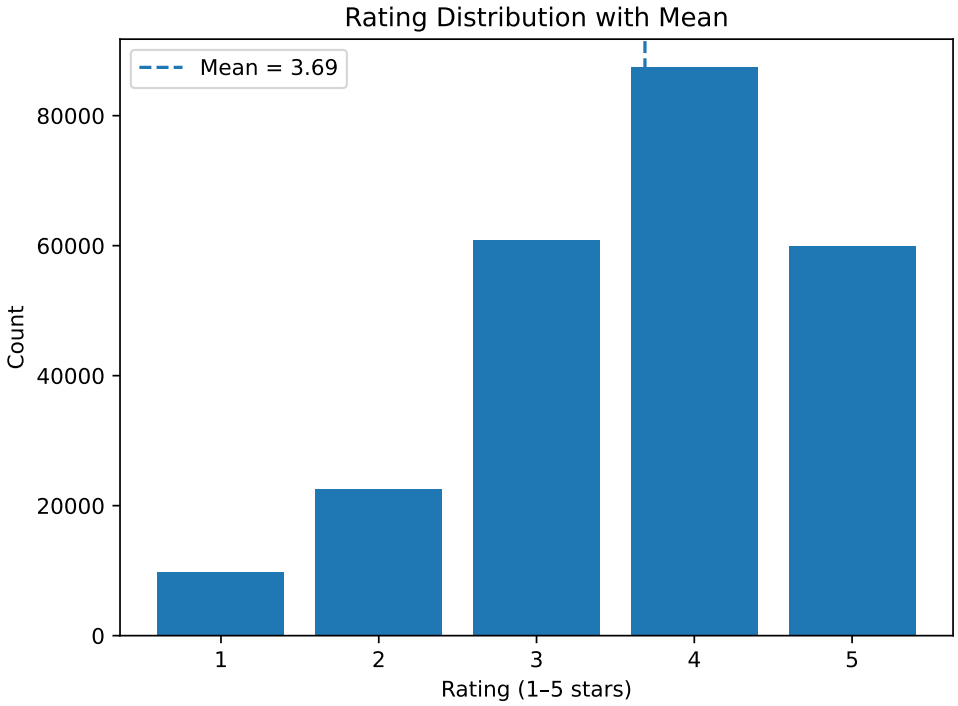

Figure 1: Dataset Dsitribution

Methodology

When we first applied a standard grid search with k-fold cross-validation to an MLP regressor, the results were unsatisfactory: due to the heavily skewed rating distribution, the model consistently predicted values near the mean (around 3.8), performing poorly on lower ratings (1–2 stars). We experimented with oversampling, undersampling, and logarithmic transformations of the target variable, but each approach degraded performance by distorting the underlying data.

Turning to techniques discussed in class, we evaluated both collaborative filtering (via matrix factorization) and content-based filtering independently. Collaborative filtering was implemented using scikit-learn's matrix factorization tools, while content-based filtering involved constructing user profiles based on genre preferences and computing Jaccard similarities between user profiles and movies, which is basically a dot product. Individually, neither method outperformed the baseline regressor.

After training both mentioned models on the test dataset, we created a new dataset with the predictions of these models. The newly formed dataset consists of the predictions on each movie/user pair from the training set, and the true rating. We then trained a classifier (normal mlp classifier) on this dataset, also using grid search and k-fold validation. Once the final model, which was a combination of the two previous models, was trained, we got an accuracy of 88%.

Finally, we explored genetic-algorithm based hyperparameter optimization for the MLP classifier. We used PyGAD. PyGAD is an open-source Python library for building the genetic algorithm and optimizing machine learning algorithms. PyGAD supports different types of crossover, mutation, and parent selection operators. PyGAD allows different types of problems to be optimized using the genetic algorithm by customizing the fitness function. It works with both single-objective and multi-objective optimization problems. Despite its computational expense, the GA failed to produce meaningful improvements over the grid-search tuned model.

Discussion

Being that the ratings are discrete classes from 1-5, thus making this a classification problem, we used accuracy as our evaluation metric and not Root Mean Squared Error, which is for regression problems.

Conclusion

We demonstrated a successful hybrid recommendation approach treating rating prediction as classification, yielding 88% accuracy. Our work highlights the value of combining collaborative and content-based signals.