Movie Recommender System

CS 487 – Final Project - Report

Devon Miller

Kitt Phi

Motivation:

Much of the success of modern streaming services comes from the implementation of recommendation algorithms. The movie industry is highly saturated, and therefore it can be very difficult for less popular movies to be discovered by audiences, even when viewers might have a strong liking for the movie's content. If streaming services can effectively recommend fresh and relevant content to users, then users are more likely to stay on the service and watch the content being provided to them, thereby increasing the revenue of subscription-based services (or other services which make profit solely off of advertising).

Given that many variables are in play when evaluating a viewer's content preferences, the mission to create a "perfect" recommendation algorithm is continuously ongoing. Any research and analysis in this area is useful, which is why we are hoping to create our own implementation to gain a better understanding of the problem and gain more experience for future work in the field.

Problem Definition:

Most modern streaming services, such as Netflix and Hulu, use machine learning algorithms to attempt to find similarities between movies based on a wide assortment of criteria. These criteria include, but are not limited to, genre, word frequency in titles, and release date. Additionally, most services track statistics such as rating and number of views. Combining these criteria allows services to quantitatively evaluate the best recommendations for users based on their previous watch history, thus improving usability and user experience.

Our goal is to create a simple movie recommender system using data from the Movie Lens dataset, one of the largest and most widely used datasets designed for this purpose. Our initial prototype will be based on item-based similarity, using both ratings and genre as a metric to evaluate the correlation between movies.

A user-based similarity approach is possible, but requires the logging of user-based information, and is therefore less in line with our scope. Designing the initial prototype with a smaller scope in mind allows us to be more flexible and augment the system later without worrying about leaving the system incomplete and non-functional.

If our work timeline allows, we plan on potentially incorporating more metrics into our recommender system, in addition to rating and genre. The main challenge with this project is that many implementations of recommender systems already exist, so our goal is to try and incorporate as many different item-based metrics as we can in order increase the robustness of our system. Finding metrics which correlate with each other and provide useful information for predicting similarities between movies will be one of the hardest aspects of this project.

Solution Explanation:

Our solution is to build an item similarity-based recommender system, which will store an index of when two people watch the same movie. When this occurs and depending on the score, the system can recommend an item to the other user because it detects that those two users are similar in terms of the movies they watched. The dataset we are using can be found at:

https://grouplens.org/datasets/movielens/. The specific dataset we are using for preliminary testing is mllatest-small.zip.

The plots show us a relationship between the number of ratings and the ratings for the movies. It shows that there is a positive relationship between the number of ratings and the average rating of a movie. Also, the graph indicates that the more ratings a movie has, the higher its average rating becomes. This causes a problem, because for the movies with few ratings but high ratings, it could allow them to be recommended even though they are not similar. For that reason, we must create a threshold when comparing the movies. By viewing the histogram of Fig. 1 we can see that a majority of the ratings per movie are under 100. Thus, only search for movies that have at least 100 reviews and then sort them ascendingly by correlation.

Data Description:

In this assignment we used the Movie Lens Dataset. The dataset was built by a research group at the University of Minnesota. It contains 100,000 ratings applied to 9,000 movies by 600 users. The data set will change over time and is not appropriate for reporting research results.

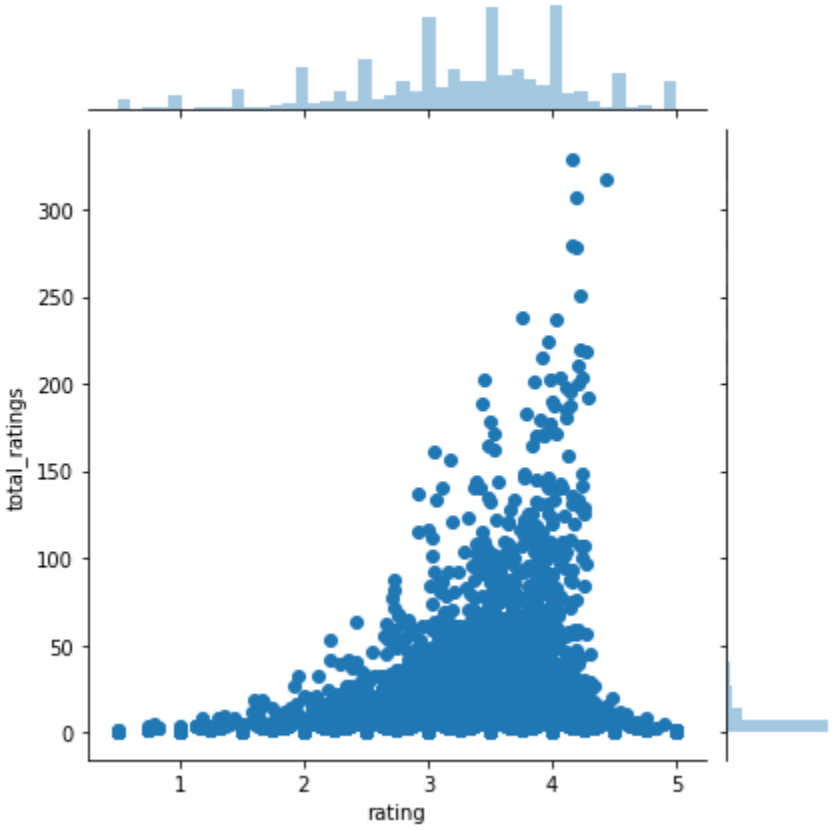

Fig.1

From looking at Fig. 1 we can see that most of the films are rated between 2.5 to 4.

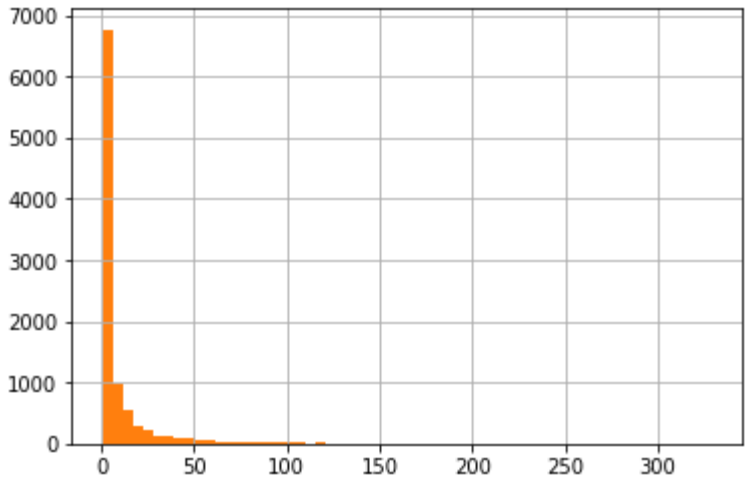

Fig. 2

From looking at Fig. 2 we can see that the that most of the movies have a small amount of ratings. The movies with the most ratings are the most famous films.

Result Analysis:

We used a dataset to find the number of ratings and the average rating for each movie. We then used the ratings to find a correlation between the movies. We used a correlation function to indicate how the two variables fluctuate together. The movies with a high correlation are the movies that have the most similarity with one another. The correlation numbers lie between –1 and 1. -1 Indicates a strong negative correlation, zero indicates no correlation (not similar at all) and 1 indicates a positive correlation. The most similar movies to Forrest Gump was Good Will Hunting; with the highest correlation of 0.48, Aladdin and American History X. The most similar movies to Pulp Fiction was Fight Club, with the highest correlation of 0.54, Kill Bill 1 and Trainspotting.

```
Top 3 most recommended movie out of 100,000 films most similar to
Forrest Gump
                          Correlation  total_ratings
title
Good Will Hunting (1997)     0.484042            141
Aladdin (1992)               0.464268            183
American History X (1998)    0.457287            129


Top 3 most recommended movie out of 100,000 films most similar to
Pulp Fiction
                         Correlation  total_ratings
title
Fight Club (1999)           0.543465            218
Kill Bill: Vol. 1 (2003)    0.504147            131
Trainspotting (1996)        0.437714            102
```