Movie Rating Prediction Analysis

CIS 563 Project Report

Lichen Liang

Abstract

With the increasing popularity of short videos in the past few years, people are diverting their attention away from long movies. In addition, the ongoing pandemic is limiting people from visiting the cinema. As a result, more focus is now put on online streaming. However, time is scarcer now than before. If one decided to watch a movie means he/she must allocate at least two hours of his/her own time. Consequently, one will prefer to watch a 'good' rated movie otherwise it will be considered a waste of time.

I. Introduction

There are a lot of movie recommender systems and algorithms used in different platforms that do a similar job: recommend a movie to a user that he/she have never watched but might like to watch. However, most recommender systems face the same problems such as cold start, data sparsity, privacy, etc. Without previous knowledge or data, it is crucial to find a way to recommend a movie to a person.

The idea of this project is to find which factors in a movie production process affect the audience the most in choosing a movie to watch. In other words, we want to analyze the data and find what factors of a movie will allow a person to choose one movie over another. For example, the factors can be a movie's director, producer, main actors, genre, plot, etc. These factors will indicate a person's thought process that leads to a good-rated movie. It also allows us to predict a movie's rating before it has been released to the public or before a large number of reviews can reflect the actual movie rating. For example, if the same director and actors produced two movies of the same series, it is expected that their third movie will get a rating similar to their first and second movie. Also, a movie's rating might not be accurate if it is released recently and has a low number of reviews. Vice versa, a movie with a lot of reviews and released long ago will likely be more accurate.

II. Prior Works

There are many types of research on recommender systems that try to solve some existing hardships using different approaches. [1] stated that despite algorithms and size of

datasets, they are based on the hypothesis that the user's opinion or review, in this case, the data does not change over time. This assumption cannot be used everywhere because we know that human behavior does change over time. As a solution, they implemented a TimeFly Algorithm that divides the classification into cycles based on timestamps. In [2], they implemented a collaborative filtering algorithm that measures the correlation between movie genres and classifies them. Lastly, recommend to users with similar taste using traditional collaborative filtering algorithm. This solves the problem of misprediction caused by recommending movies to similar-minded users from user-based collaborative filtering. In [3], they used user based collaborative filtering by creating two lists that the user likes and dislikes. From these two lists, then recommend movies based on the positive and negative user profiles.

III. Data and Model

The data used in this project is very different from other data found online. It contains 28 features that contain the compositions of the movie. A full description of features is shown in table 1.

Table 1.

Feature	Description
color	Color of the movie (black and white / colored)
director_name	Name of the director of the movie
director_facebook_likes	Number of Facebook likes of this director
actor_1_name	Name of main actor
actor_2_name	Name of supporting actor
actor_3_name	Name of supporting actor
actor_1_facebook_likes	Number of Facebook likes of this main actor
actor_2_facebook_likes	Number of Facebook likes of this supporting actor
actor_3_facebook_likes	Number of Facebook likes of this supporting actor
gross	Gross revenue of this movie
num_critic_for_reviews	Number of critical reviews
duration	Duration of the movie
genres	Genres of the movie, may include multiple values

movie_title	Title of the movie
num_voted_users	Number of users who voted this movie
cast_total_facebook_likes	Total number of likes this cast got on Facebook
facenumber_in_poster	Number of actors shown on movie poster
plot_keywords	Keywords of the movie, may include multiple values
movie_imdb_link	A URL link to this movie on IMDb
num_user_for_reviews	Number of users who wrote a review for this movie
language	Main language used in movie
country	Country of this movie was produced
content_rating	Content rating of this movie
budget	Budget of this movie
title_year	Year this movie was released
imdb_score	IMDb score of this movie
aspect_ratio	Aspect ratio of this movie
movie_facebook_likes	Number of likes this movie got on Facebook

Data preprocessing

There are some features that should not be evaluated or are not relevant to the final prediction. In other words, a movie should not be rated based on such features. These features include color, movie's IMDb link, language, country, movie title, and content rating. Therefore, these features are dropped from the dataset. There are also missing values in the dataset. The rows containing missing values are also dropped.

In order to run the classification models, all the data should be numerical. For each categorical feature, for example, director name 'James Cameron', replace this name with its value count in that feature. Assuming the name 'James Cameron' directed 7 movies, then all the instances containing this name will be changed to 7 as an integer. Such categorical features include director name, actors' names, genres, and keywords. Instead of directly replacing with values, a new feature is created containing the value, merged to the data, and categorical columns are dropped.

Classification Models

There are four regression models used: Linear, Lasso, Ridge, and Random Forest along with other classification methods: Support Vector Machine (SVM), K-Nearest Neighbor (KNN), and Gaussian Naïve Bayes (GNB). The data is first converted from type 'float' to type 'int' and split into train and test of size 90% and 10% respectively. The feature that needs to be predicted is imdb_score.

IV. Analysis and Results Discussion

The rooted mean square errors of the regression models are shown in table 2. The accuracy of SVM, KNN, and GNB are shown in table 3.

Table 2.

Model Name	RMSE
Linear	0.871
Lasso	0.881
Ridge	0.871
Random Forest	0.685

Table 3.

Model Name	Accuracy
SVM	0.385
KNN	0.542
GNB	0.209

We can see that the Linear, Lasso, and Ridge regression have very similar rooted mean squared errors. Whereas Random Forest regression achieved the lowest error rate. KNN achieved the highest accuracy and Naïve Bayes has lowest accuracy.

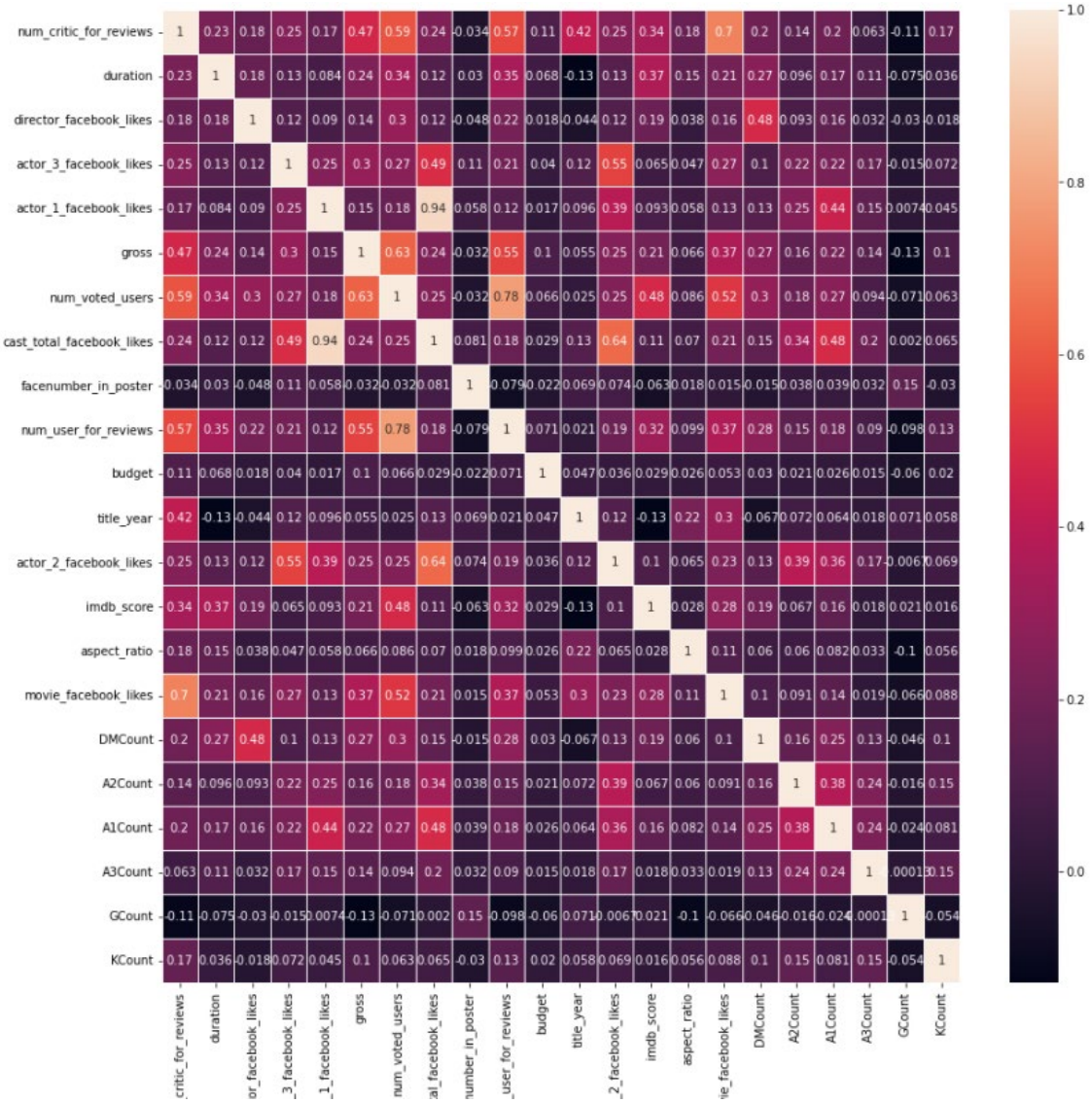

Figure 1. Correlation heatmap

From figure 1, we can find the correlations to imdb_score. The top five features with highest correlations are num_voted_users, duration, budget, num_critic_for_reviews, and num_user_for_reviews. This is somewhat expected because the imdb score in general is directly affected by user reviews. For example, a user leaving a critical review is more likely to rate a lower rating and vice versa. For budget, a movie with higher budget is more likely to be better than ones with lower budget. However, for duration of a movie, it is questionable whether it plays an important role in affecting the rating of a movie. It is also surprising that the directors and actors are not as correlated to the rating.

V. Conclusion and Future Improvement

There are some procedures that can be implemented to increase the accuracy of the models and decrease the errors. For example, the dataset is relatively small, and we have removed many rows that contained null values. One improvement is to replace null values using mean. Moreover, the genres and keywords features are composed of multiple values, we can further decompose and analyze them. Many features consist of Facebook likes, it faced the same problem of static data mentioned in [1], so we should update the values and even get more data and ratings from different sites.

VI. References

[1] Arun Kumar R., Deepika R., Sathesh Kumar K., Dinesh P. S. (2021). Recommender System : An Automated Movie Rating Prediction Using an Improved Timefly Algorithm (ITFA). *Annals of the Romanian Society for Cell Biology*, 1967–1971.

[2] Hwang, T.-G., Park, C.-S., Hong, J.-H., & Kim, S. K. (2016). An algorithm for movie classification and recommendation using genre correlation. *Multimedia Tools and Applications*, *75*(20), 12843–12858. https://doi.org/10.1007/s11042-016-3526-8

[3] Chen, Y.-L., Yeh, Y.-H., & Ma, M.-R. (2021). A movie recommendation method based on users' positive and negative profiles. *Information Processing & Management*, *58*(3), 102531. https://doi.org/10.1016/j.ipm.2021.102531