Movie Recommendation System

Neel Popat
B.Tech Ahmedabad University
Ahmedabad, India
neel.p@ahduni.edu.in

Kashish Jivani
B.Tech Ahmedabad University
Ahmedabad, India
kashish.j@ahduni.edu.in

Yashvi Navadia
B.Tech Ahmedabad University
Ahmedabad, India
yashvi.n@ahduni.edu.in

Abstract

Nowadays, OTT platforms serve an ample amount of movies from various genres. So to make the users' experience better, we propose the idea of making a movie recommendation system based on a classical machine learning approach. The recommendation can be provided using content based filtering and collaborative filtering.

Keywords

Recommendation Systems, Cosine Similarity, Weighted Mean, Singular Value Decomposition (SVD), Machine Learning

Introduction

A recommendation system will be made using ML techniques like weighted mean, cosine similarity function, etc. Firstly, a Simple recommender will be made which will provide recommendations based on the votes received by the users. Later on, content based and collaborative based filtering will be made which provides personalized recommendation to users. Finally, a hybrid recommender which will be a mixed version of content based and collaborative based filtering.

Literature survey

- Research about the various techniques which can be applied to build a recommendation system for end users.
- The pros and cons of every module used for recommendation.

Implementation

The movie recommendation system is currently divided into:

A. <u>Simple recommendation system</u>

Simple recommendation system works on the basis of weighted rating from the dataset. The weighted mean is calculated from the below formula.

Weighted Rating

$$wr = \frac{v}{v+m} * R + \frac{m}{v+m} * C$$

where:

v is the number of votes for the movie

m is the minimum votes required to be listed in the chart

r is the average rating of the movie

c is the mean vote across the whole report

The recommender will work on the basis of the genre input provided and the audience total ratings and total rating count of the movie.It will be general to every user and not personalized based on the user's previous watchlist.

B. Content based filtering

The two content based engines are build that

- one that took movie overview and taglines as input
- the other which took metadata such as cast, crew, genre and keywords to come up with predictions.

We also devised a simple filter to give greater preference to movies with more votes and higher ratings. Here, the Cosine Similarity is used to calculate a numeric quantity that denotes the similarity between two movies. Mathematically, it is defined as follows:

$$cosine(x, y) = \frac{x.yT}{||x||.||y||}$$

The dot product returns the cosine similarity score directly. The matrix will be generated which will contain the pairwise cosine similarity matrix for all the movies in our dataset. Then we return the 30 most similar movies based on the cosine similarity score.

C. Collaborative based filtering

There are two types of collaborative filtering: user collaborative filtering and item collaborative filtering. This type of filtering works on the basis of past views of users i.e. if user A has rated one movie high and another low, and user B has the same kind of rating, then it can be inferred that they are alike. So recommending each other's movies can be a very good idea.

D. Hybrid Recommendation system

The hybrid system brings together the content based system and collaborative filtering which makes use of input of User Id and title of movie and the model returns the sorted movies on the basis of expected ratings by the particular user. If we look in particular for the hybrid recommender we can notice that for our hybrid recommender, we get different recommendations for different users although the movie is the same. Hence, our recommendations are more personalized and tailored towards particular users.

Results

The results from the simple recommender system returns the same result for every genre input provided irrespective of the user demanding the recommendation till date. The results for the content based filtering are yet to be confirmed. The expected result for content based filtering will require comparing movies with their cast, genre, etc which will match the movies with most of the similarities and the output will be the list of movies as suggestions to watch in future since they liked a particular/set of movies in the past based on similar cast and crew, genre, keywords, etc.

Conclusion

Based on the initial results, the users will get the benefit to find movies based on various genres they like. Adding to that, they will also enjoy movie recommendations with similar casts and keywords. For recommendations to be close to what users will like, our sample dataset should be as close as the population dataset. Since we are using a small dataset, the error may vary for various inputs and it is likely that the model at this phase will need more training in order to lessen the real time loss of users in terms of false recommendations.

References

a) A. Mukherjee, “Building a recommendation system using weighted-average score,” *Medium*, 16-Sep-2019. [Online]. Available: https://medium.com/@developeraritro/building-a-recommendation-system-using-weighted-hybrid-technique-75598b6be8ed. [Accessed: 19-Mar-2022].

b) Rounak Banik, “Movie Recommender Systems,” Kaggle, 06-Nov-2017. [Online]. Available: https://www.kaggle.com/code/rounakbanik/movie-recommender-systems/notebook. [Accessed: 19-Mar-2022].

c) Ibtesama, “Getting started with a movie recommendation system,” Kaggle, 16-May-2020. [Online]. Available: https://www.kaggle.com/code/ibtesama/getting-started-with-a-movie-recommendation-system/notebook. [Accessed: 19-Mar-2022].