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 divided into:

A. Simple recommendation system

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

Weighted Rating

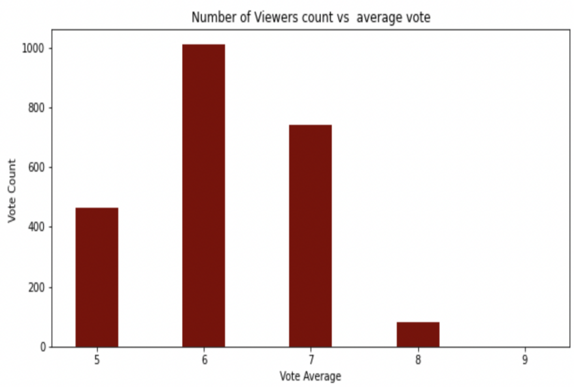
$$wr = \frac{v}{v+m} * R + \frac{m}{v+m} * C$$

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.



Vote Average Vs Vote Count

Weighted mean based filtering

Sr. No.	title	vote_count	vote_average	Weighted rating
10309	Dilwale Dulhania Le Jayenge	661	9	8.565285
351	Forrest Gump	8147	8	7.971357
876	Vertigo	1162	8	7.811667
40251	Your Name.	1030	8	7.789489
883	Some Like It Hot	835	8	7.745154

B. <u>Content based filtering</u>

This recommender is divided into two parts based on the metadata of movies:

- one that takes movie overview and taglines as input
- the other which takes cast, crew, keywords and genre as input

TF-IDF vectorizer is used to transform text into vectors as quantitative data isn't available. Here, the Cosine Similarity is used to calculate a numeric quantity that denotes the similarity between two movies. Mathematically, it is defined as follows:

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

Since TF-IDF is used, the dot product returns the cosine similarity score directly. The matrix generated contains the pairwise cosine similarity score for all the movies in our dataset. We also devised a simple filter to give greater preference to movies with more votes and higher ratings. 10 most similar movies based on the cosine similarity score are displayed.

Content based filtering - Overview, tagline

10 recommendations on “The Dark Knight”

Sr.No	Movie
7931	The Dark Knight Rises
132	Batman Forever
1113	Batman Returns
8227	Batman: The Dark Knight Returns, Part 2
7565	Batman: Under the Red Hood
524	Batman
7901	Batman: Year One
2579	Batman: Mask of the Phantasm
2696	JFK
8165	Batman: The Dark Knight Returns, Part 1

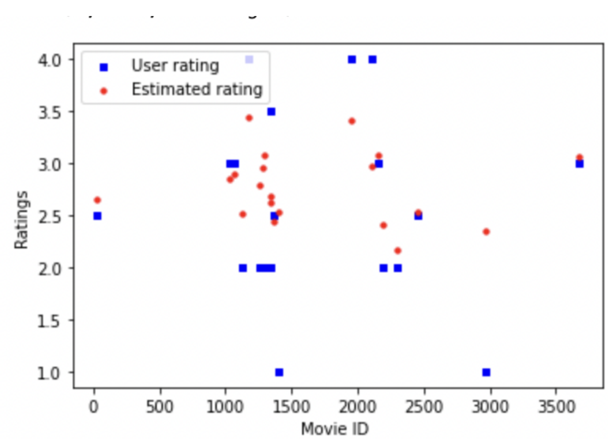
Content based filtering - Cast, crew, keywords, genre

10 recommendations on “The Dark Knight”

Sr.No	Movie
8031	The Dark Knight Rises
6218	Batman Begins
6623	The Prestige
2085	Following
7648	Inception
4145	Insomnia
3381	Memento
8613	Interstellar
7659	Batman: Under the Red Hood
1134	Batman Returns

C. <u>Collaborative based filtering</u>

This filtering provides personalized recommendations to the users. It works on the basis of ratings given by different users to the movies. “Surprise” library is used for implementing this algorithm. SVD is used to minimize Root Mean Square Error and give great recommendations. It works purely on the basis of an assigned movie ID and tries to predict ratings based on how the other users have predicted the movie.



Movie ID Vs Ratings

D. <u>Hybrid Recommendation system</u>

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.

1) 10 recommendations on movie Avatar for user 1

Id	Title	Estimated ratings
218	The Terminator	3.083605
280	Terminator 2: Judgment Day	2.947712
127585	X-Men: Days of Future Past	2.935140
18887	Darby O'Gill and the Little People	2.899612
679	Aliens	2.869033
54138	Star Trek Into Darkness	2.806536
16306	Fantastic Planet	2.789457
2756	The Abyss	2.774770
5227	Hercules in New York	2.703766
25628	Hawk the Slayer	2.680591

2) 10 recommendations on movie Avatar for user 500

Id	Title	Estimated ratings
54138	Star Trek Into Darkness	3.238226
679	Aliens	3.203066
14164	Dragonball Evolution	3.195070
14821	Escape to Witch Mountain	3.149360
14822	Return from Witch Mountain	3.138147
597	Titanic	3.110945
280	Terminator 2: Judgment Day	3.067221
127585	X-Men: Days of Future Past	3.043710
218	The Terminator	3.040908
16306	Fantastic Planet	3.018178

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 expected result for content based filtering will require comparing movies with their tagline, 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. The result tables of each recommender system are shown above.

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) C. Z. Omega and Hendry, "Movie Recommendation System using Weighted Average Approach," 2021 2nd International Conference on Innovative and Creative Information Technology (ICITech), 2021, pp. 105-109, doi: 10.1109/ICITech50181.2021.9590147.

b) S. Agrawal and P. Jain, "An improved approach for movie recommendation system," 2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), 2017, pp. 336-342, 10.1109/I-SMAC.2017.8058367.

c) C. M. Wu, D. Garg and U. Bhandary, "Movie Recommendation System Using Collaborative Filtering," 2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS), 2018, pp. 11-15,