



MB Recommendations - An OTT platform for movies and books

Omkar Chinchkar¹, Anvesha Kulkarni², Vrushali Dudhe³,
Shreyash Lokhande⁴, Dr. Nikita Kulkarni⁵

Computer Engineering, Trinity Academy of Engineering, Pune, India

I. ABSTRACT

As we all know in today's world, the usage of the internet has increased which directly leads to an increase in data. So finding the exact content that the user wants is getting harder. This is why personalized recommendation systems have become a necessity in many areas such as entertainment, e-commerce, social media, etc. Recommendation Systems are designed to make the user experience much better. This system will help users to save their time and increase the probability of user's satisfaction as a result; users will purchase more content, so the service provider will also earn more. That's why, we are making a recommendation system which will recommend movies and books to users as per their preferences. Now while creating a recommendation system we should take note of the fact that how unique are the choices of each individual in terms of movie preferences. Some people like genrespecific movies such as a thriller, romance, or sci-fi, while others focus on lead actors, directors and production houses. When we take all that into account, it's very difficult to generalize a movie or a book and say that everyone would like it. But with all that said, it is still seen that similar movies are liked by a specific part of the society, so the system will make the group of users with similar taste. For example, if one user from one group likes a horror movie there is a large possibility that other members from the same group will like the same horror movie as well. For this, various methods such as content-based filtering, collaborative filtering are used.

Keywords: Recommendation system, Content-based filtering, Collaborative filtering, Books recommender.

II. INTRODUCTION

A recommendation engine is a type of data filtering engine that uses machine learning algorithms to recommend the most relevant items for a specific use. It works on the principle of looking for patterns in consumer behavioral data, which can be collected implicitly or explicitly.

As we all know in today's world, the usage of the internet has increased which directly leads to an increase in data consumption. So finding the exact content that the user wants is getting harder. This is why personalized recommendation systems have become a necessity in many areas such as entertainment, e-commerce, social media, etc. Recommendation Systems are designed to make the user experience much better. This system will help users to save their time and increase the probability of user's satisfaction as a result; users will purchase more content, so the service provider will also earn more. But on the other hand, it may look relatively simple but behind the scenes, a complex statistical algorithm gets executed in order to predict these recommendations. Recommendation system is one of the most famous applications of data science and machine learning. As a result, we developed a movie recommendation system that will suggest movies or books to consumers based on



their tastes. Now, when constructing a recommendation system, we need to consider how distinctive each person's movie interests are. Talking about the movies some people prefer genre films, such as thrillers, romances, or science fiction, while others are more interested in the leading stars, directors, and production companies. This system also recommends books based on the user's movie taste[9]. With all of this in mind, it's tough to generalize a film and declare that everyone will enjoy it. Even though it is clear that identical films are enjoyed by a specific segment of the population, the system will create a group of users with similar tastes [2]. For example, if one user from one group likes a horror movie there is a large possibility that other members from the same group will like the same horror movie as well. For this, various methods such as content-based filtering, collaborative filtering are used. Each method has several advantages over another method so we will use both of these techniques to make recommendations better.

Content-Based recommender system attempts to guess the features or behavior of the item's features given by a user, if a user reacts positively to author used features such as movie plots, ratings, countries of production and release year to find similarity between movies [8]. Collaborative Filtering is used to predict the rating of items. For every user Collaborative filtering method uses a user-item-rate matrix. We are going to use Kaggle MovieLens movies Dataset for movies and Kaggle Goodreads Books Dataset for books in this system.

III. LITERATURE SURVEY/BACKGROUND WORK

In this recommendation system, data is filtered according to the user choice. Basically, the study of recommender systems started in the early 90's and has increased over the time. In these studies, various powerful methods have been designed and implemented. Those methods are content-based filtering, collaborative filtering. In recent years, deep learning and reinforcement learning methods have also been used [2].

● Used Techniques

1. Collaborative Filtering:-

The primary use of collaborative filtering is to predict the rating of an item. For each individual user, a "user-item-rate matrix" is used for collaborative filtering methods. Each column and row of this matrix indicates a specific item and user. Collaborative filtering method is divided into two main subgroups: model-based method and memory-based method. In model-based methods, the model first learns and then recommends. The matrix factorization is the most popular method to reduce the number of unknown parameters, it factorizes the user - item-rate matrix into two low-rank matrices. The matrix factorization is the most common way to reduce the number of unknown parameters. It factorizes the user-item-rate matrix into two sub-matrix [4][5]. One of the major problems that recommendation systems face is the cold start problem. The main reason behind sparsity is that most users do not provide ratings and those available are often too scattered or rare; for example rating of any mobile app. Most of the time users ignore when asked to rate an app, and even when users do, they rate either positive or negative, so the available ratings are quite scattered. This is a big problem for collaborative filtering where recommendations are based on user ratings [3].

2. Content-Based Filtering:-



Content-based filtering uses item characteristics and user action history to learn about user preferences. So, in this method, we train the model by determining the characteristics of each element. Content-based systems work on the concept of similar content that users have liked in the past. The basic idea behind such a system is to recommend similar items or products that have been enjoyed by users in the past. Based on similar characteristics, the degree of similarity between two or more items can be calculated. For better understanding, continuing with the example, whenever you watch an Instagram video, like the one posted on Animal Lovers page, when finished watching that video, you will get the similar videos while scrolling on your page. Also, when you like certain pages/sites, you'll get recommendations of similar sites. So what's behind all of this is content-based filtering i.e. if you like an item from a particular category, chances are you want any other similar items from the same category [8].

3. Hybrid Method:-

It is the combination of two filtering methods i.e. content-based and collaborative filtering. The idea is to combine the features of two methods in such a way that it gives the best of both worlds. The rate estimation of an item and a user can be achieved in this method by the rating pattern of the similar users to the same item. For our recommendation system we are gonna be using this hybrid method [1].

● Algorithms

1. K-Nearest Neighbor(KNN):-

To find similar users/items memory-based methods use the KNN algorithms. K-nearest neighbor (KNN) is a popular classification method in data mining and statistics because of its simple implementation and significant classification performance. It is based on supervised learning techniques. The KNN algorithm stores all available data and ranks a new data point based on similarity. This means that when new data comes in, it can easily be classified into a good category using the K-NN algorithm. K-NN algorithm is divided into two types i.e. Regression as well as for Classification but mostly it is used for the Classification problems [6].

IV. PROPOSED WORK/SYSTEM

1. USE CASE DIAGRAM



Fig.3 Use Case Diagram

2. SYSTEM ARCHITECTURE

When a new user wants to use a system, the user has to sign up and provide some explicit feedback to the system (for ex. genres that the user is interested in or the languages that the user is familiar with). Then the system will recommend content according to user’s feedback. The system will also recommend the latest movies and popular movies.

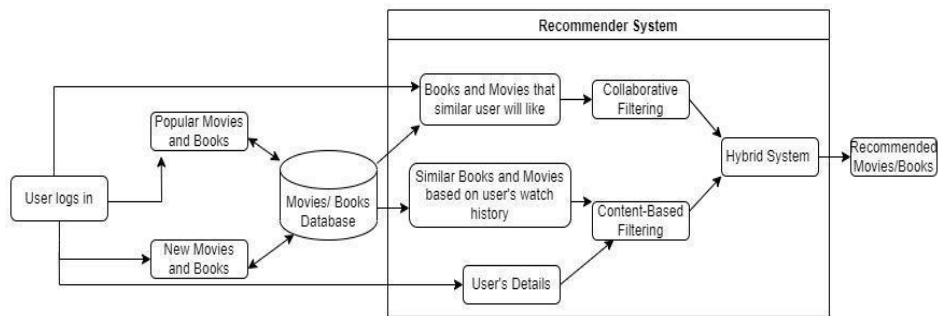


Fig. 4 System Architecture

A collaborative filtering system makes the clustering of similar users (They may like similar genres, similar actors, similar authors) so when one user likes some specific content that content will also be recommended to other users from same cluster(group).

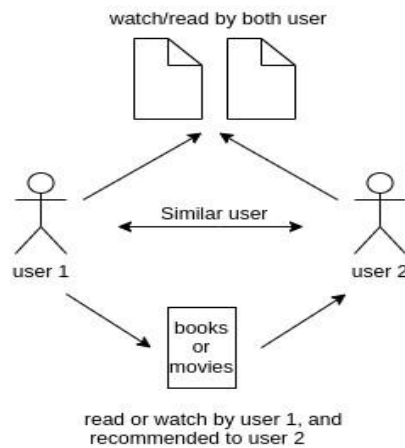


Fig 4.1 Collaborative Filtering

A content based filtering recommends the content according to the user's past choices. It monitors the user's watch history and recommends accordingly. It also uses the explicit data provided by the user for better understanding of the user's choices.

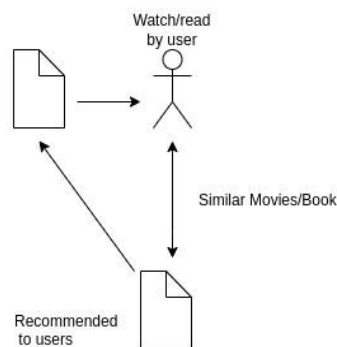


Fig 4.2 Content-based Filtering



3. TECHNOLOGIES USED

a. Python

Python offers concise and readable code. While complex algorithms and versatile workflows stand behind machine learning, Python's simplicity allows developers to write reliable systems. Developers get to put all their effort into solving an ML problem instead of focusing on the technical nuances of the language.

b. React JS

React.js is a JavaScript library that is used for building UI/UX used for single-page applications. Web page made using React.js gives an app-like experience in the browser as it is a single page application.

c. Node JS

Node.js is an asynchronous event-driven JavaScript runtime used to create REST-API. Node.js applications never buffer any data. These applications simply output the data in chunks. It also has **Express JS** as a framework which makes it easy to develop web applications fast.

d. HTML

HTML (HyperText Markup Language) is the code that is used to structure a web page and its content. HTML structure editor, providing structure for all the page's components, including its header element, footer element, main content, and other inline elements.

e. CSS

CSS (Cascading Style Sheets) is used to add design to a website by dictating how a site is displayed on a browser. CSS is unique in that it doesn't create any new elements, like HTML or JavaScript. Instead, it's a language used to style HTML elements.

f. JavaScript

JavaScript is a scripting language that runs in the browser (also in your local environment with NodeJS). JS is behind every action/interaction you have performed on the browser. With JS you can create web apps that will run on any machine with a browser.

g. Surprise

Surprise is a Python module that makes it simple for programmers to create recommender systems using explicit rating data.

h. Numpy

NumPy may be used to conduct a wide range of array-based mathematical operations. It extends Python with sophisticated data structures that ensure fast computations with arrays and matrices, as well as a large library of high-level mathematical functions that work with these arrays and matrices.

V. RESULTS

1. MOVIE PAGE

In this module we are showing movie recommendations related to the current(now playing) movie and also books recommendations related to current(now playing) movies. So that the user will not only get more movies related to the topic of the movie he/she is watching but also more book to read as well on same topic.

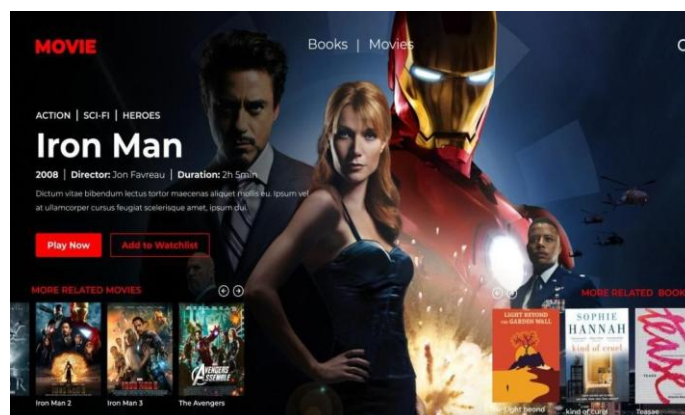


Fig. 3 Movie page

2. PERSONALIZED RECOMMENDATIONS

Here there are two different modules one for books and another for movies, both give personalized top-n recommendation respectively. This module will first get all explicit as well as implicit data of the user and recommend movies and books to the user in their respective page.

```
We recommend:  
For userID no.:- 238 :  
Sleepers (1996) 5  
Beavis and Butt-Head Do America (1996) 5  
Fugitive, The (1993) 5  
In the Line of Fire (1993) 5  
Mrs. Doubtfire (1993) 5  
Home Alone (1990) 5  
Flintstones, The (1994) 5  
Twister (1996) 5  
Princess Bride, The (1987) 5  
Summer of Sam (1999) 5
```

Fig. 4 User recommendations

3. ACCURACY

The most important aspect of the recommendation system is its accuracy. It can be evaluated through two main measures: Mean Absolute Error and Root Mean Squared Error is a measure of how spread out these residuals are. It tells you; how concentrated the data is around the line of best fit. Mean Absolute Error is an average of the absolute difference between the predicted rating and the actual rating. They are negatively-oriented scores(0-∞), Lower values better the prediction [7].

Actually this doesn't calculate the accuracy of recommendation, instead it calculates the error by a module compared to train set data and test set data. Lower the value more accurate the predictions.

- a. **Mean Absolute Error** : MAE is the average of the absolute differences between predictions. Evaluation and actual evaluation



$$MAE = \frac{\sum_{i=1}^n |y_i - x_i|}{n}$$

- b. **Root Mean Square Error** : RMSE is the standard deviation (prediction error) of the residuals. Residual is a measure of how far a data point is from the regression line.

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (y_i - x_i)^2}{n}}$$

In our module we got more accurate results as shown in figure below. We got less errors in predictions compared to another algorithm

Algorithm	RMSE	MAE
ContentKNN	0.9375	0.7263
Random	1.4385	1.1478

Fig. 5 Accuracy Error Result

VI. CONCLUSION

Our system uses user ratings to recommend movies and books. The genre of the movie/book, the topic being discussed in the book/movie, and characteristics of actors, authors, etc. are also considered as proposals. This approach provides lower RMSE and MAE values than other conventional systems. Other More Efficient Comparison Methods Provides the accuracy of the recommendation system, such as hit ratio, average mutual hit ratio (ARHR), and cumulative hit ratio promising results. Other important aspects such as coverage, diversity and novelty. A list of unique films Books made for individual user recommendations are being improved. In your system the recommendations made for every individual user are having less error.

Our system is helpful when a user wants to read more about the movie he/she watched or watch the movie on a topic that he/she read. For example, if the user is watching a movie called ‘The Tashkent Files’, a 2019 Indian Hindilanguage thriller film about the death of former Prime Minister of India, Lal Bahadur Shastri. Then user will get real time book suggestions of books like ‘Your Prime Minister is Dead’ by Anuj Dhar, ‘Lessons in Leadership’ by Pavan Choudary, ‘The Tashkent Files’ by Mr. Vivek Kumar Pandey Shambhunath. Here this gives an all new experience to watch movies as well as for reading books.



REFERENCES

- [1] Nguyen N.T., Rakowski M., Rusin M., Sobiecki J., Jain L.C. (2007) Hybrid Filtering Methods Applied in Web-Based Movie Recommendation System. In: Apolloni B., Howlett R.J., Jain L. (eds) Knowledge-Based Intelligent Information and Engineering Systems. KES 2007. Lecture Notes in Computer Science, vol 4692. Springer, Berlin, Heidelberg. Rahul Kataria, Om Prakash Verma. An effective collaborative movie recommender system with cuckoo Issue 2, July 2017, Pages 105-112.
- [2] Mishra N., Chaturvedi S., Mishra V., Srivastava R., Bargah P. (2017) Solving Sparsity Problem in Rating Based Movie Recommendation System. In: Behera H., Mohapatra D. (eds) Computational Intelligence in Data.
- [3] 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, doi: 10.1109/ICSESS.2018.8663822. <https://ieeexplore.ieee.org/document/8663822>
- [4] M. K. Kharita, A. Kumar and P. Singh, "Item-Based Collaborative Filtering in Movie Recommendation in Real time," 2018 First International Conference on Secure Cyber Computing and Communication (ICSCC), 2018, pp.340-342, doi: 10.1109/ICSCCC.2018.8703362 <https://ieeexplore.ieee.org/document/8703362>
- [5] S. Zhang, X. Li, M. Zong, X. Zhu and R. Wang, "Efficient KNN Classification With Different Numbers of Nearest Neighbors," in IEEE Transactions on Neural Networks and Learning Systems, vol. 29, no. 5, pp. 1774-1785, May 2018. doi: 10.1109/TNNLS.2017.2673241. <https://ieeexplore.ieee.org/document/7898482>
- [6] Chai, Tianfeng & Draxler, R.R.. (2014). Root mean square error (RMSE) or mean absolute error (MAE)?—Arguments against avoiding RMSE in the literature. Geoscientific Model Development.7. 1247-1250. 10.5194/gmd7-1247-2014.
- R. Singla, S. Gupta, A. Gupta and D. K. Vishwakarma, "FLEX: A Content Based Movie Recommender," 2020 International Conference for Emerging Technol(INCET), 2020, pp. 1-4, doi: 10.1109/INCET49848.2020.9154163. <https://ieeexplore.ieee.org/document/9154163>
- [7] M. Kommineni, P. Alekhya, T. M. Vyshnavi, V. Aparna, K. Swetha and V. Mounika, "Machine Learning based Efficient Recommendation System for Book Selection using User based Collaborative Filtering Algorithm," 2020 Fourth International Conference on Inventive Systems and Control (ICISC), 2020, pp. 66-71, doi:10.1109/ICISC47916.2020.9171222. <https://ieeexplore.ieee.org/document/9171222>
- [8] https://www.amazon.in/gp/product/9386473356/ref=as_li_qf_asin_il_tl?ie=UTF8&tag=bookscharmi05-2021&creative=24630&linkCode=as2&creativeASIN=9386473356&linkId=9237e91848149facd153c03324ef2ded
- [9] https://www.amazon.in/gp/product/9380710364/ref=as_li_qf_asin_il_tl?ie=UTF8&tag=bookscharmi05-2021&creative=24630&linkCode=as2&creativeASIN=9380710364&linkId=92683379f5aaeea7b41fa116e85dc0c9
- [10] https://www.amazon.in/gp/product/1646784758/ref=as_li_qf_asin_il_tl?ie=UTF8&tag=bookscharmi05-2021&creative=24630&linkCode=as2&creativeASIN=1646784758&linkId=647bb8ff41f02ae4641ac4308b9ff891



[13] <https://www.zee5.com/movies/details/the-tashkent-files/0-0-73415>

[14] <https://www.marvel.com/explore>