iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** V **Month of publication:** May 2023

DOI: https://doi.org/10.22214/ijraset.2023.53204

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue V May 2023- Available at www.ijraset.com

Implementing a Movie Recommendation System in Machine Learning Using Hybrid Approach

Prof. Rupali Jadhav[1], Snehal Inamdar[2], Sakshi Karanjkar[3], Ritesh Karankal[4], Riya Kate[5]
Department of Computer Engineering, Zeal College of Engineering and Research, Pune

Abstract: ***In this paper, we have proposed a movie recommendation system using hybrid recommendation system. Today, there are a lot of recommendation systems available which are practically implemented in various websites and mobile apps. Variety exists in types of recommendation systems, user interfaces but most importantly, the accuracy of the recommendation systems. Determining a user's possible future preference of movie or TV shows to watch is a complex task which requires a lot of relevant user data such as watch history of user, genres liked by the user, favorite actor or director, etc. Hence, the aim of this proposed system is to refine the search engine and make it more enhanced and accurate in terms of prediction. The system recommends the movies graphically based on both, user preference and similarity of individual user with other users. It also shows top rated movies worldwide and updates the recommendation after every choice of movie or show by the user.***

Keywords: Machine Learning, Recommendation System, Algorithms, Demographic Filtering, Content Based Filtering, Collaborative Filtering, Hybrid Filtering

I. INTRODUCTION

One type of information filtering system that sorts items based on user interests is a recommendation system. The problem in overload in recent decades has made the recommendation system an essential component of e-commerce websites and social media. Recommendation systems have emerged in an age of information overload to help users find interesting items based on their preferences or choices. It was used in a variety of fields, including e-commerce and online learning. Examples of such applications include Movie Lens, Netflix, YouTube, Amazon.com, Facebook, and others evaluation of Basic Concepts. Many such information filtering and recommendation systems exist but the success of such a system is measured by its accuracy that is, how accurately the machine learning model can group similar users, calculate and process user ratings and reviews and recommend movies in a way user is more likely to select it and stream the movie or the show.

II. EXISTING SYSTEM

A hybrid recommendation system works on the core idea of combining multiple recommendation approaches like Demographic Filtering, Content-based Filtering, Item-based and User-based Filtering and Collaborative Filtering approaches. The Machine Learning model is trained based on the past or the historic data and then provides accurate recommendation on the sample or input data. Matrix factorization and clustering model are similar approaches used in techniques which are model-based. SVD which is Single Value Decomposition is a popular M.F. technique that has gained significance in the Netflix price challenge' s system of recommendations. The existing system works on a single users' rating measure and on another parameter that is the genre of the show.

III. PROPOSED SYSTEM

In proposed system, a combination of demographic, content-based and collaborative filtering in mainly used.

A. Demographic Filtering Approach

The meaning of the word demographic is the part of statistics that describe certain properties of data and their characteristics such as age, gender, etc. Various movie recommendation systems use demographic filtering approach. This approach makes use of the demographic information of system users to recommend them items matching their profile, preferences, user history, etc. The main perk of demographic filtering approach is that it uses IMDB weighted rating algorithm or formula to calculate movie ratings. E.g. Suppose a movie has received 8.2 rating but only has 5 votes. Such a movie cannot be considered a good movie as compared to the one that has got 7.8 rating with 15 votes. Hence, it is evident that number of users influences the rating of the movie. Hence, the IMBD weighted rating formula is industry standard to determine movie ratings worldwide.

©IJRASET: All Rights are Reserved

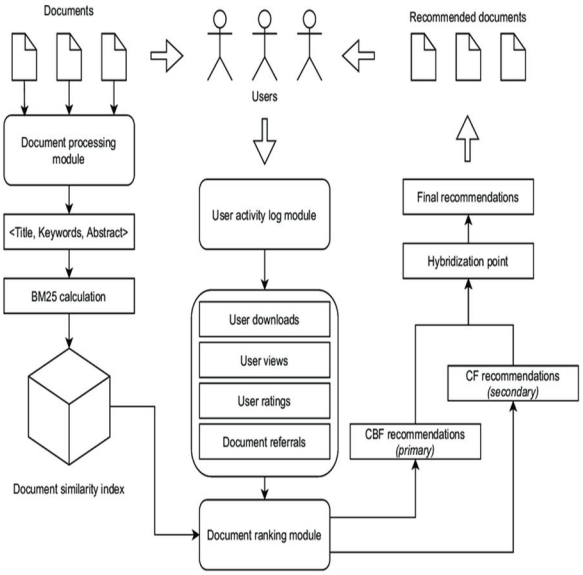
B. *Content-Based Filtering*

Content-Based filtering approach in movie recommendation system uses individual user's previous watch history and recommends the movies or TV shows accordingly. It focuses on the user's likes and dislikes, favorite genre, favorite directors or actors. This information is processed by the machine learning model and the movies or shows the user is most likely to watch in the future are recommended based on this data. The recommended results, if streamed by the user are also added to the existing user information to refine the recommendation process. Content-Based approach uses algorithms like Cosine Similarity and TF-IDF (Term Frequency- Inverse Document Frequency).

C. *Collaborative Filtering*

Collaborative filtering approach is the one which combines or pair multiple users together based on similarity between their preference or watch history to recommend the movies. E.g. If a movie 'X' is a Sci-Fi, Action movie watched by two users 'A' and 'B'. These two users will be paired together just because they watched the above-mentioned movie. User A might have watched it because A might like Sci-Fi movies. On the other hand, B must've watched it because B likes Action movies. Also, since A and B are now paired together, movies watched by A but not by B will be recommended to B. Same goes for user A. Collaborative approach uses algorithms like Pearson Correlation, Matrix Factorization (MF) and Single Value Decomposition (SVD).

IV. SYSTEM ARCHITECTURE



V. SYSTEM IMPLEMENTATION

The successful implementation of any machine learning system or model depends upon the quality and integrity of the data used. In this system, TMDB dataset is used which is the standard dataset for movie recommendation systems. Basic data preprocessing techniques are applied and the model is trained on the data. A user interface (UI) is made which will help backend communicate with the user. The UI, which is the front-end part is a website made using HTML, CSS and Javascript. At the backend, when model is finished training, it is connected to the front-end part using Express. Express is a NodeJS framework containing multiple ports and inbuilt functions that effortlessly connect backend to it's compatible front-end. The client-side communicates with server-side using HTTP requests. The task of grouping similar users is achieved by using Pearson Correlation to mathematically group the users in various categories to ease and proceed with the recommendations. After the models such as collaborative and content-based filtering models are developed and evaluated for performance, a hybridization approach is used to combine the recommendations from both the models. Weighted averaging is a common hybridization technique where recommendations from both the sides are merged together depending upon their respective accuracies.

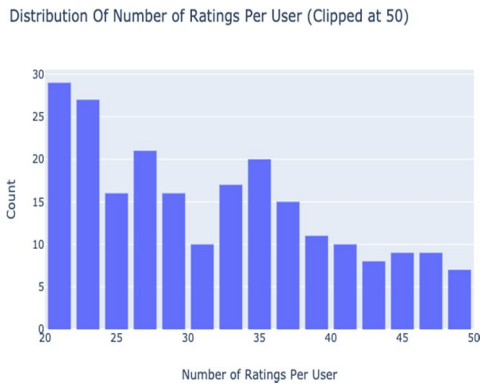
To integrate the recommendation system that uses TMDB API for recommendations into the MERN app, we need to obtain the API key for TMDB in order to gain access the movie datasets and get recommendations for multiple users. The Express.js backend of

the MERN app needs to be updated to handle requests and responses between the frontend and backend. The frontend of the MERN app needs to upgraded to include a recommendation feature that can send user preferences to the backend and receive recommendations based on them. The MongoDB database should store user and movie data, which will be used to generate recommendations.

As for API design and communication, the API design should be RESTful, using HTTP methods like GET, POST, PUT, and DELETE to handle requests and responses. The API should communicate between the frontend and backend, sending recommendations to users based on their preferences. The frontend should send a request to the backend API with the user's preferences, and the backend should use the TMDB API to retrieve a list of recommended movies based on those preferences. The recommended movies can be filtered based on the user's viewing history and rating history to provide a personalized recommendation experience. Additional features can also be added at the frontend part i.e. UI or the website such as a chatbot for related help, a featured cast list of the movies or TV shows under the trailer iframe of the movie which will be further aided by Google Lens to redirect users to a Google search query on their favorite actors, latest news about them and their upcoming new movies and shows.

VI. CONCLUSION

There are several datasets available to build a movie recommendation system. But for this project, we are going to use a dataset that contains the metadata (cast, crew, budget, etc...) of the movie. The best dataset for this is the TMDB database namely TMDB 5000 dataset. This project uses an algorithm for hybrid filtering recommendation system and we have applied it in the movie recommendation system. This personalized recommendation system uses the singular value decomposition algorithm and User- based cosine similarity algorithm; these recommend the top n movies to the active User. Along with these, other algorithms such as Pearson Correlation and Matrix Factorization have also been used to enhance the performance of the machine learning recommendation model.



REFERENCES

[1] Davidsson C, Moritz S. Utilizing implicit feedback and context to recommend mobile applications from first use.DOI: 10.1051/,04008 (2017) 712012ITA 2017 ITM Web of Conferences itmconf/201 40084 In: Proc. of the Ca RR2011. New York: ACM Press, 2011. 19-22.http://dl.acm.org/citation.cfm?id=1961639[doi:10.1145/1 961 634.1961

[2] Bilge, A., Kaleli, C., Yakut, I., Gunes, I., Polat, H.: A survey of privacy-preserving collaborative filtering schemes. Int. J. Softw. Eng. Knowl. Eng. 23(08), 1085–1108 (2013) CrossRefGoogle Scholar.

[3] Calandrino, J.A., Kilzer, A., Narayanan, A., Felten, E.W., Shmatikov, V.: You might also like: privacy risks of collaborative filtering. In: Proceedings of the IEEE Symposium on Security and Privacy, pp. 231–246, Oakland, CA.

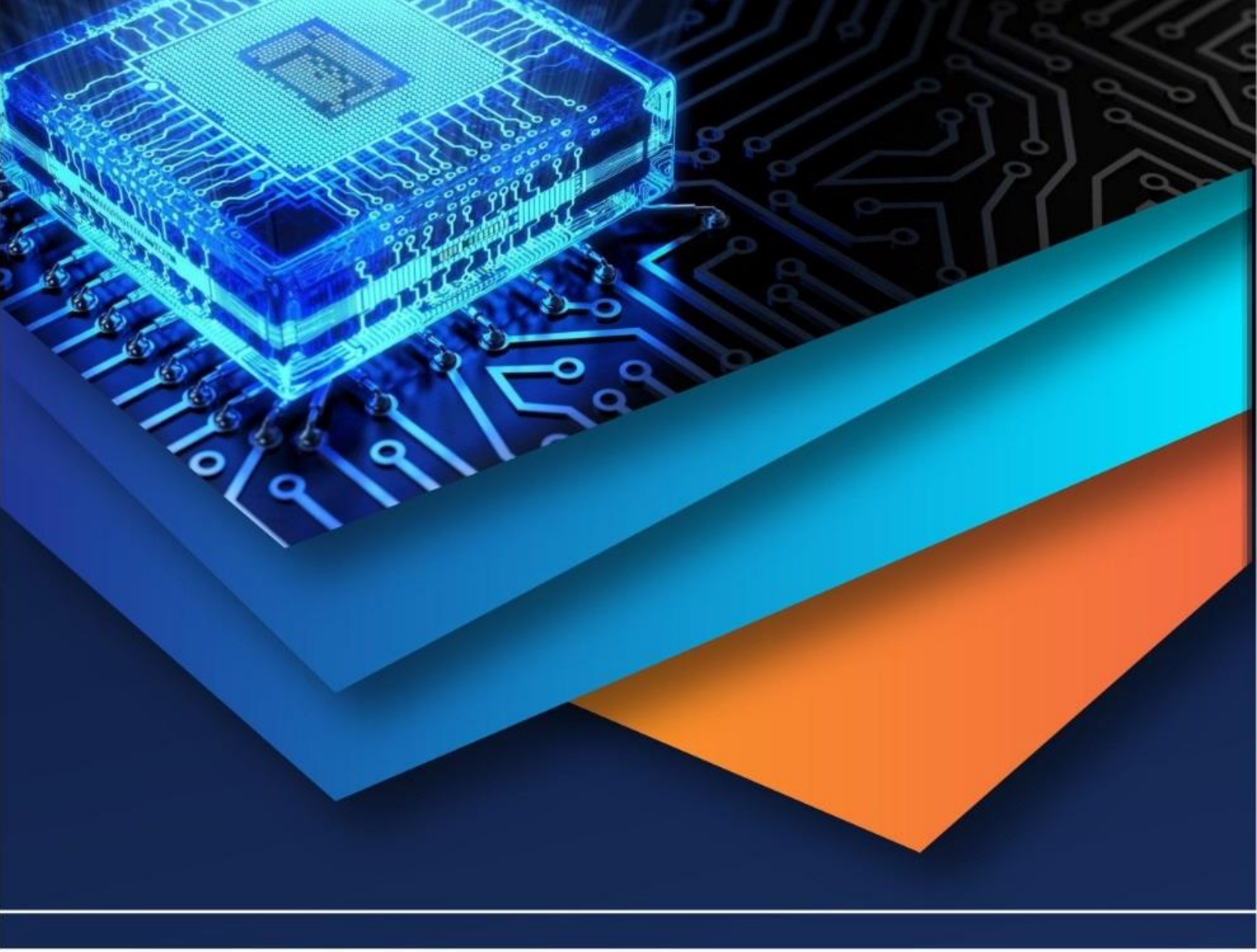
[4] Okkalioglu, M., Koc, M., Polat, H.: On the discovery of fake binary ratings. In: Proceedings of the 30th Annual ACM Symposium on Applied Computing, SAC 2015, pp. 901–907. ACM, USA (2015).

[5] A Hybrid Approach for Movie Recommendation System Using Feature Engineering by S. Sathiya Devi and G. Parthasarathy.

[6] A Comprehensive Survey on Movie Recommendation Systems by R. Lavanya, Utkarsh Singh, Vibhor Tyagi.

[7] Fully Content-Based Movie Recommendation System With Feature Extraction Using Neural Network by Hung-Wei Chen, Yi-Leh Wu, Maw-Kae Hor., Cheng-Yuan Tang.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)