



e-ISSN: 2319-8753 | p-ISSN: 2347-6710

IJRSET

International Journal of Innovative Research in
SCIENCE | ENGINEERING | TECHNOLOGY



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 12, Issue 5, May 2023

ISSN INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA

Impact Factor: 8.423

9940 572 462

6381 907 438

ijirset@gmail.com

www.ijirset.com

Movie Recommendation System

Aniket Kakekar¹, Atish Kasar², Roshan More³, Aditya Elalkar⁴, Ankush Hutke⁵

U.G. Student, Department of Information Technology, Rajiv Gandhi Institute of Technology (Mumbai University),
Mumbai, Maharashtra, India^{1,2,3,4}

Asst. Professor, Department of Information Technology, Rajiv Gandhi Institute of Technology (Mumbai University),
Mumbai, Maharashtra, India⁵

ABSTRACT: Recommendation systems have become increasingly important in various fields with the explosion of big data, such as e-commerce, social networks, and web-based services. Among various recommendation techniques, Collaborative Filtering (CF) is one of the major and prevailing techniques for implementing recommendation systems. CF methods predict user preferences based on the preferences of similar users, and have been shown to provide improved performance metrics such as Root-Mean-Square Error (RMSE) in movie recommendation applications, where publicly available movie datasets such as MovieLens and Netflix are widely used. Despite the success of CF methods in improving recommendation performance, two major issues are still being faced by movie recommendation systems. The first issue is scalability. Traditional CF methods require significant computational resources, making them infeasible for large-scale datasets. The second issue is practical feedback and verification based on real-world implementation. Many recommendation systems often suffer from the cold-start problem, where the system lacks sufficient user data to provide accurate recommendations.

KEYWORDS: Recommendation, Preferences of users, Major issues in recommendation.

I. INTRODUCTION

A Recommender System, also known as a recommendation engine, is a type of filtering system designed to anticipate an individual's preference or rank for a particular item. These systems are mainly used for commercial purposes and are widely employed in industries such as entertainment and e-commerce. Recommender networks are frequently used to create personalized playlists and video suggestions on platforms such as Netflix, YouTube, and Amazon Music, as well as product recommendations on websites such as Amazon. These systems can be designed to work with a single input, such as music, or multiple inputs, such as books, news, and search queries, both within and across platforms. Online recommendation systems play a critical role in helping users discover relevant information. To achieve this, movie recommender websites frequently ask users to provide feedback in the form of ratings to improve the recommendation process. Online movie recommender systems like IMDB and MovieLens suggest favorite movies to users based on their movie ratings and watchlist.

II. PROPOSED SYSTEM

We proposed One-hot encoding technique to recommend the movies and cosine similarity to finding the similarity.

- One-hot conversion: - It is a technique used to represent categorical data in machine learning algorithms. It represents each category as a binary vector where each dimension of the vector corresponds to a unique category. In order to calculate similarity between two binary vectors that have been encoded using one-hot encoding.
- Cosine similarity: - We are going to use cosine similarity metric. The cosine similarity metric measures the cosine of the angle between two vectors. It is a measure of the similarity between the two vectors, where a value of 1 indicates that the vectors are identical, and a value of 0 indicates that the vectors are completely dissimilar. To calculate the cosine similarity between two vectors encoded using one-hot encoding, we are following these steps:
 1. Multiply the two vectors element-wise.
 2. Sum the resulting values.
 3. Calculate the magnitudes of each vector.



4. Multiply the magnitudes of the two vectors.
5. Divide the sum of the element-wise products by the product of the magnitudes.

In one-hot encoding, each dimension of the binary vector represents a unique category, and its value is either 0 or 1. The dot product of two binary vectors equals the number of matching dimensions, and the magnitude of a binary vector equals the square root of that number. According to their genre, we determine the similarity or dissimilarity between two vectors encoded using one-hot encoding by measuring their cosine similarity.

III. METHODOLOGY

The proposed methodology offers an effective approach to building a genre-based movie recommendation system that can provide accurate and personalized recommendations to users based on their movie preferences.

Following are the steps to build the system: -

1. Data Collection and Pre-processing: We have collected the data from Kaggle, Movie lens and from the websites like IMDB and Wikipedia by web scraping technique. Collected data was very inconsistent and lot of missing values so we used different pre- processing techniques and inbuilt python libraries like pandas and NumPy.
2. User Input: We take the input from the user what movie they like. User can select the movie based on their genre from the database we have
3. Filtering: We have filtered out those movies who have poor rating or genre which not more likely watched by the different users by analysing the data using different filtering techniques
4. One-Hot conversion: we have defined the function which converts the data into a binary vector format which is called one hot conversion.
5. Making data frame for all bit values: created the data frame in which all rows represent the movie titles, rating with their genre and all columns represent the unique genres presented in the dataset. Here if specific genre is presented in respective movie then data that genre bit value should be 1 otherwise it should be zero.
6. Similarity calculation: when user enter a movie the algorithm calculates the similarity with the other movies in the dataset using bit vectors and cosine similarity and gives the result and sorted the list with rating.
7. Recommendation Generation: Similar movies are organized in the manner such that the movie with highest rating will be at the top of list.
8. Presentation: Presenting the recommended movies to the user in the list format along with their rating and cast also. We are presenting the movies based with posters.

IV. EXPERIMENTAL RESULTS

A user-friendly interface is created as shown in Fig 1 to register for the site which will allow the user to access the Movie Recommendation Engine. A user-friendly interface refers to a design that is easy to use and understand for the user and understand for the user. It is intuitive and straightforward, allowing the user to accomplish tasks quickly and efficiently without feeling overwhelmed. A good user interface should have clear navigation, concise labeling, and visual cues to guide the user's actions. The interface should be straightforward, allowing the user to accomplish tasks quickly and efficiently without feeling overwhelmed. A good user interface should have clear navigation, concise labeling, and visual cues to guide the user's actions. The interface should also be responsive, meaning it should provide immediate feedback to the user's actions. In doing so, a user-friendly interface should enhance the user's experience and make their interaction with the system or product as seamless as possible.

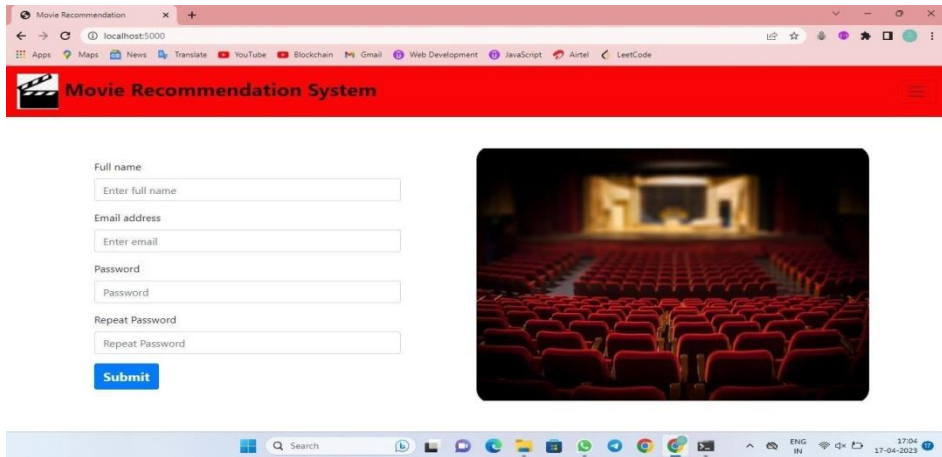


Fig 1: User Interface for registration of new account.

Once a user has registered for the site, they can access it anytime from the login page as shown in Fig 2. The Login Page prompts users to authenticate their identity before accessing the service. The login page requires users to provide their username or email address and password to gain access. The design of the login page is simple, clear, and easy to use to ensure that users can quickly and easily authenticate their identity.

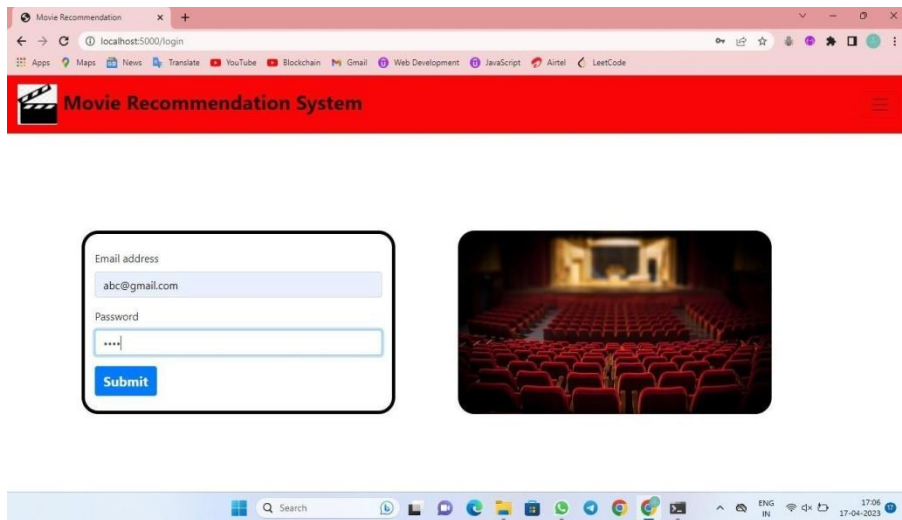


Fig 2: LoginPage.

Once the user has completed their login they can finally search for their desired move. The users will be greeted with a movie search engine shown in Fig 3. The users can enter their query into the search bar to get appropriate recommendations. Here for example, The Amazing Spider- Man was searched as illustrated in Fig 4. The result shows the movie title, movie poster, a short summary of the movie and the cast details.

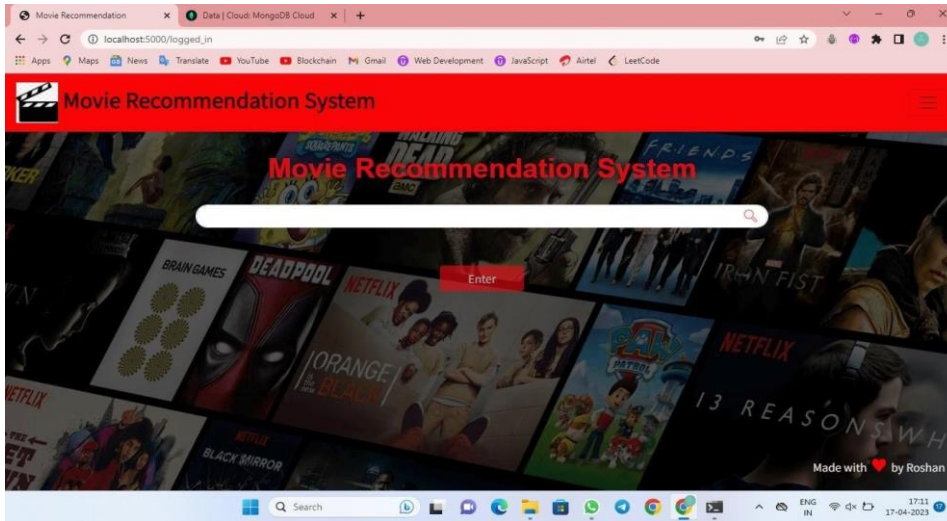


Fig 3: Movie Search Interface.

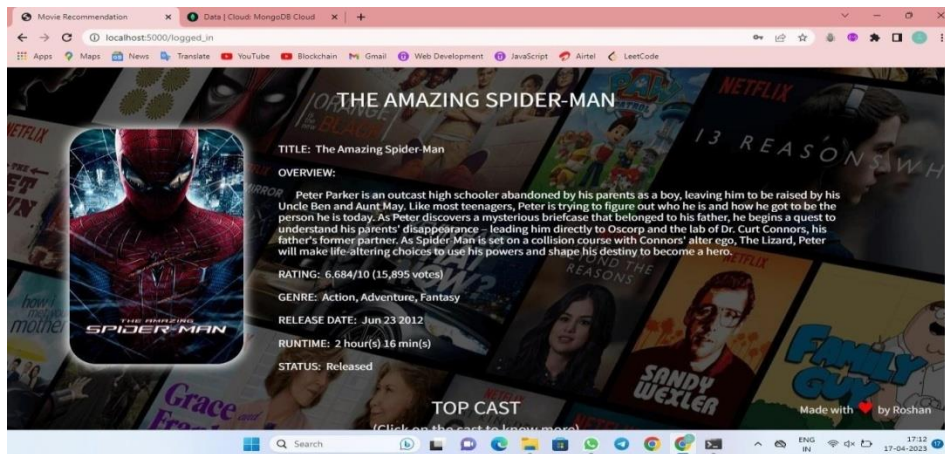


Fig 4: Search Result for the desired query.

With the help of the recommendation system implemented, the user will also find other recommendations similar to the query they have searched for. The results for the query we used as an example are shown in the Fig 5 below.

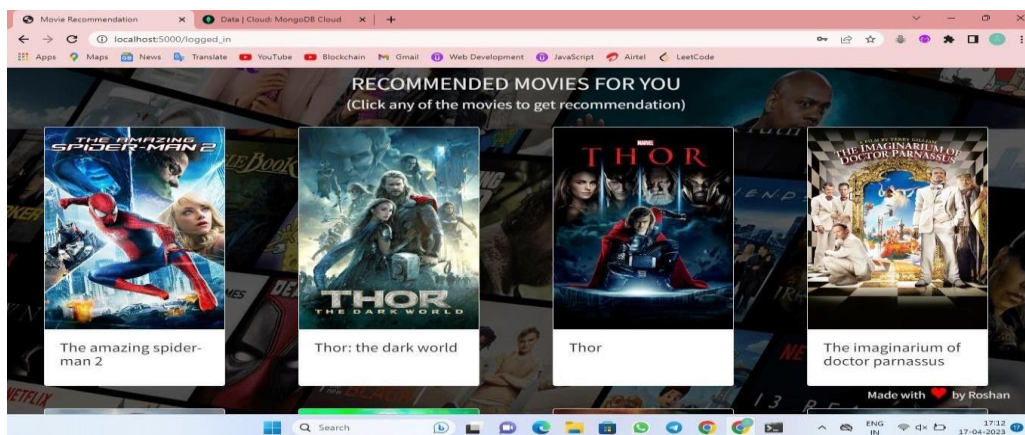


Fig 5: Recommendations similar to the searched query.



V. FUTURE SCOPE

Even with all the benefits added by the proposed system, there is always room for more enhanced experience. This research allows for some issues which are needed to be addressed in future for a more fulfilling experience. They are as follows:

- Addressing the cold start problem: Collaborative filtering systems often struggle to provide accurate recommendations for new users who have not yet provided enough ratings to generate meaningful recommendations. Future research will focus on developing more effective techniques to address the cold start problem, such as using content-based filtering or hybrid recommendation systems.
- Incorporating social media networks: Social-media can provide a wealth of information about users, including their preferences and social connections. This would be a much better approach for profile building. Incorporating social network information into collaborative filtering systems will help tremendously for a more accurate and relevant movie recommendation.
- Integrating contextual information: The Proposed model do not consider contextual information such as the time of day or the user's location. Incorporating such contextual information could help improve the relevance and accuracy of recommendations.
- Exploring user-item interactions beyond ratings: While user ratings are a common source of data for collaborative filtering, there are other ways in which users interact with movies that can be leveraged for recommendation purposes. For example, users may write reviews, add movies to their watchlists, or click on related content. Incorporating these interactions could provide additional insight into user preferences and help improve recommendation accuracy.

VI. CONCLUSION

This system is a genre-rating based recommender system which is based on Collaborative Filtering. Collaborative filtering is a powerful technique used in movie recommendation systems that aims to predict a user's preferences based on their historical behavior and similarities with other users. By analyzing patterns in user ratings and identifying correlations with other users' preferences, the system can generate personalized movie recommendations that are highly likely to appeal to the user's taste. This approach has proven highly effective and popular due to its reliance on real-world data and user behavior, which makes it adaptable to a range of different data sets and contexts. One of the key advantages of collaborative filtering is its scalability and flexibility, as it can handle large volumes of data and provide personalized recommendations for millions of users and movies. Additionally, it does not require any explicit knowledge of the movies or the users, making it a highly versatile technique that can be applied to a range of different industries and domains.

REFERENCES

1. M. S. Faisal, A. Rizwan, K. Iqbal, H. Fasihuddin, A. Banjar and A. Daud, "Prediction of Movie Quality via Adaptive Voting Classifier", doi.10.1109/ACCESS.2022.3195228.
2. J. Zhang, Y. Wang, Z. Yuan and Q. Jin "Personalized real-time movie recommendation system: Practical prototype and evaluation", doi: 10.26599/TST.2018.9010118.
3. Liu H, Hu Z, Mian A, Tian H, Zhu X. , "A new user similarity model to improve the accuracy of collaborative filtering. Knowl Based Syst.", 2014;56:156– 66. <https://doi.org/10.1016/j.knosys.2013.11.006>.
4. M. Chenna Keshava , P. Narendra Reddy , S. Srinivasulu , B. Dinesh Naik , "Machine Learning Model for Movie Recommendation System" . doi: 10.17577/IJERTV9IS040741.202
5. Camacho LA, Alves-Souza SN. "Social network data to alleviate cold-start in recommender system: a systematic review." Inf Process Manag. 2018;54:529–44. <https://doi.org/10.1016/j.ipm.2018.03.004>.



Impact Factor: 8.423



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

 9940 572 462  6381 907 438  ijirset@gmail.com



www.ijirset.com

Scan to save the contact details