IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 12, Issue 4, April 2024

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.379

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 4, April 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1204226 |

Movie Recommendation System with Hybrid Filtering

Prof. Ashwni Bhosale, Sudarshan Waghmare, Arshan Kazi, Sujit Jadhav, Nikhil Jagtap

Department of Computer Technology, Siddhant College of Engineering, India

ABSTRACT: This project offers a method for making movie recommendations. It has been a highly common method for recommending films in recent years, and many OTT platforms utilize it. The Movie Recommendation System continuously improves itself by learning from user comments and producing recommendations that become more accurate over time. It saves users time and effort by creating a personalized list of movie suggestions, introducing users to new and obscure films, and allowing the sharing and discussion of recommendations. Since movie recommendations are crucial to our social lives because they can boost our delight. A system like this can recommend a variety of films to consumers based on their interests or the popularity of a certain film.

I. INTRODUCTION

A movie recommender system is an intelligent piece of software that makes movie suggestions to users based on their tastes, past viewing habits, and other pertinent information. In order to create personalized suggestions, it analyses vast datasets, including user ratings, movie metadata, and user behavior patterns. It does this by using cutting-edge algorithms and techniques from machine learning and data mining. A movie recommender system's main objective is to improve users' movie-watching experiences by assisting them in finding movies they're likely to like. Users can rely on the system to get specialized recommendations that match their tastes and preferences rather than manually looking through a vast library of films.The system provides personalized movie recommendations by taking into account user choices and taking into account their individual inclinations. It considers elements including genre preferences, ratings, and interesting recommendations. Based on user input and interactions, the algorithm continuously learns and adjusts. The system's main objective is to make it easier for users to choose films, to do it quickly and easily, and to improve their overall movie-watching experience. The system promises to connect users with films that resonate with their interests and preferences by offering personalized recommendations and a user-friendly interface, ultimately enabling them to discover new films and experience a more fulfilling cinematic trip.
In this project, we constructed three distinct movie recommendation approaches, starting with a count vectorizer for natural language processing-based content-based movie recommendations. We utilize cosine similarity to determine how closely two films are related. Second, we had to apply a characteristic known as Pearson's correlation coefficient in our Collaborative filtering movie recommender system that used matrix factorization. We also used the "KNN" algorithm to construct a collaborative filtering recommender system
.

II. LITERATURE REVIEW

The era of information and communication technology makes the information available on the internet growing rapidly. Recommender Systems are one of the technologies that are widely used to filter information to handle a huge of information. One of the developing pieces of information is film. The increasing number of films released yearly has led to the development of applications that offer movie streaming services such as Netflix, Yiu, Disney Hotstar, etc. Therefore, movie recommender systems technology is needed to facilitate and provide a good experience when users use these services.

[1] Movie Recommendation System Jose Immanuvel. J , Sheelavathi. A , Priyadharshan. M , Vignesh. S , Elango. K, International Journal for Research in Applied Science & Engineering Technology (IJRASET), June 2022.
A movie recommendation system can be built using a variety of datasets. However, for this project, we'll use a dataset that includes the movie's metadata (cast, crew, budget, etc.). The algorithm for a collaborative filtering recommendation system used in this study was applied to the recommendation system for films. The User- based Co-Coin Similarity Algorithm and Singular Value Decomposition Algorithm are used in this personalised recommendation system to provide the active User with recommendations for the top n films.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 4, April 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1204226 |

[2] Mahesh Goyani and Neha Chaurasiya, "A Review of Movie Recommendation System: Limitations, Survey and Challenges", Electronic Letters on Computer Vision and Image Analysis 19(3):18-37, 2020 The various filtering methods are described in this study. The various applications, benefits, and drawbacks are also highlighted. A hybrid combination of many recommendation systems is necessary to create an effective recommender system. It is concluded that utilizing a combination of similarity measures will result in better user similarity than using a single measure, and the system's efficiency will also increase. One of the facts is that the author developed measurements like RJMSD, which up to this point have only been used for efficiency characteristics. Any recommender system's accuracy can be increased by including more movie features. In general, the majority of the articles have demonstrated the use of both collaborative and content-based filtering.

[3] S. K. Raghuwanshi and R. K. Pateriya, "Collaborative Filtering Techniques in Recommendation Systems," in Data, Engineering, and Applications (Springer, Singapore, 2019).
By enabling shops to offer personalized recommendations to customers based on data obtained from the Internet, recommendation systems have the ability to explore new business prospects. They assist customers in quickly locating the goods and services that closely fit their preferences. Additionally, a variety of cutting-edge algorithms have been created to suggest products depending on how customers interact with their social networks. As a result, research into integrating social media photos into systems that propose clothing has been quite popular in recent years. Based on scholarly literature on the subject, this report reviewed fashion recommendation systems, algorithmic models, and filtering strategies. The technical features, advantages, and disadvantages of the filtering algorithms have been thoroughly addressed, facilitating the in-depth comprehension of future researchers.

[4] R. Ahuja, A. Solanki, and A. Nayyar, "Movie recommender system using k-means clustering and k-nearest neighbor" in Proceeding of the 9th International Conference on Cloud Computing, Data Science andEngineering, (Confluence, 2019), pp. 263–268.
Machine learning is a method of data analysis that automates the development of analytical models. It is an area of artificial intelligence that was established on the idea that robots can learn from data, recognize patterns, and form judgments with little help from people. In the suggested system, a movie recommender system is developed using the K-Means Clustering and K-Nearest Neighbour algorithms. The data originated from the movie lens data set. Theconcept is put into practice using the Python programming language. Once the system has been programmed in Python, it can be demonstrated that the proposed technique has a lower RMSE numberthan the one currently in use.

[5] Wang, A. P. de Vries, and M. J. T. Reinders, "Unified relevance models for rating prediction in collaborative filtering," ACM Transactions on Information Systems, vol. 26, no. 3, pp. 1 –42, June 2008.
Recommender systems are proving to be a useful tool for tackling a portion of the internet's record overload issue. Its development has coincided with that of the Internet. The primary technology of the recommender system collected data from the content-based record, demographic statistics, and memory- based information using ordinary websites. Recent studies demonstrate how sentimental analysis can be used to create recommender systems that are more accurate. These kinds of techniques are frequently applied in e- commerce operations. The author of this study has categorized numerous sentiment analysis-based recommender system techniques.

[6] Paul Marx- Providing Actionable Recommendations: A Movie Recommendation Algorithm with Explanatory Capability, Joseph EulVerlag,2013.
In this paper, the author summarises the findings and consequences of the study and makes suggestions for additional research. The first portion of this chapter provides a concise summary of the actions we took to carry out our analysis and develop our method. Our contributions to the RS literature are also listed there. The second half of this chapter focuses on the main implications of our findings for the manufacturers and developers of recommendation systems. Finally, the last section of this chapter concludes our thesis by presenting potential research topics and making recommendations for how to improve our proposed recommendation procedure.

III. METHODOLOGY

After installing and importing all necessary packages, we got our dataset from Kaggle, which is labeled **"tmdb_5000_movies.csv"** for content-based filtering and **"User ratings.csv"** and **"tmdb_5000_movies.csv"** for collaborative filtering.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 4, April 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1204226 |

First, we used natural language processing to construct a **"Content-Based movie recommender system"**. A "Bag of words" is a notion used in natural language processing. Sentences in Bag of Words are altered into a group of meaningless words, whereas sentences in real life are changed into fixed-length vectors of numbers. Because of this,each word in this model gets a special number that represents how frequently it has been used. Next, we focus on how the word is represented rather than the word order.

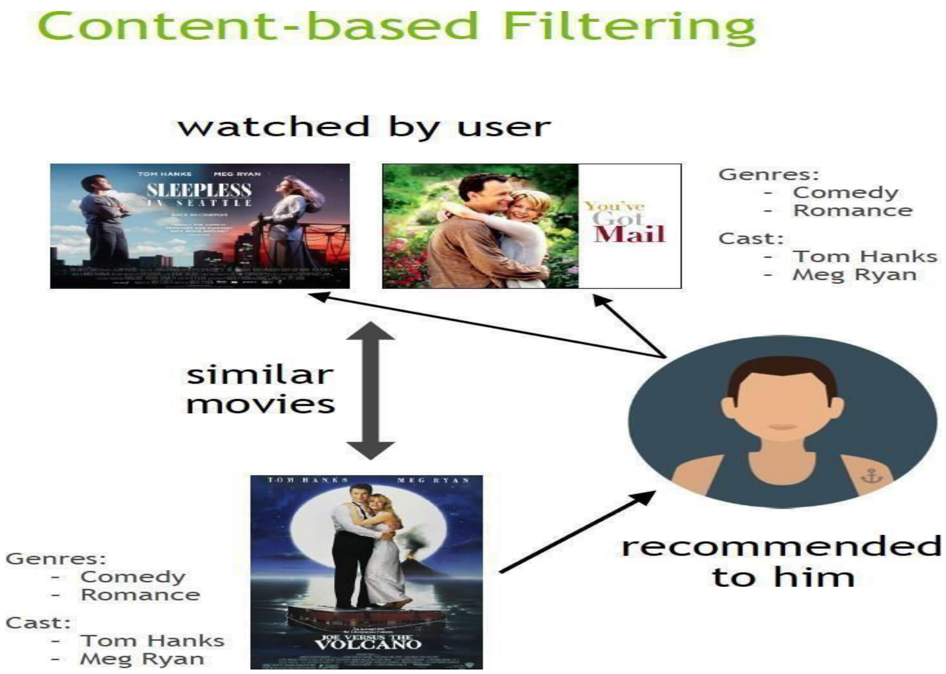

Fig 1: An explanation picture of how a content-based recommender system works

For the content-based recommendation system, **"count vectorization"** was applied. A common method in natural language processing (NLP) for turning a collection of text documents into numerical feature vectors for use in machine learning methods is count vectorization. A sparse matrix of counts is used to indicate the frequency of words or concepts in a document.

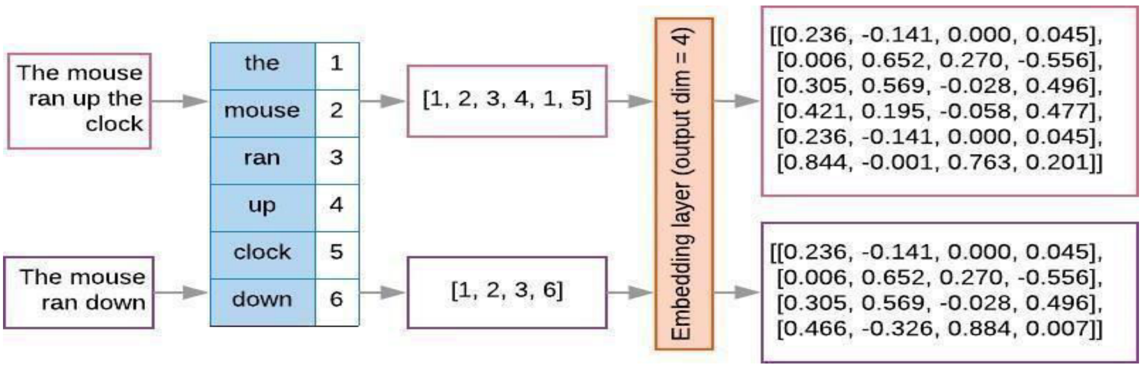

Fig 2: Explanation of how a count vectorizer works.

We'll utilize **"Cosine-similarity"** to determine how closely two films are related. The cosine similarity index determines how similar two vectors are to one another in an inner product space. By determining the cosine of the angle between two vectors, it can determine if they are roughly pointing in the same direction. It is commonly used in text analysis to determine how similar two documents are.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 4, April 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1204226 |

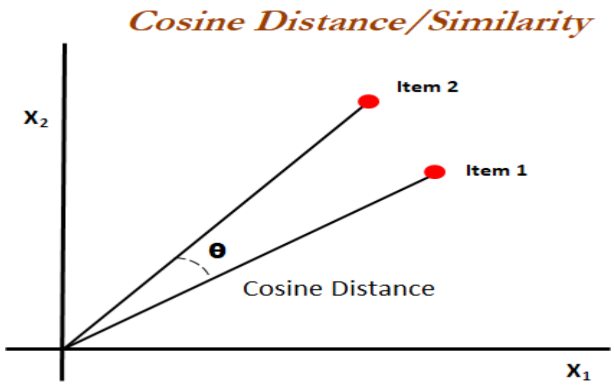

Fig 3: A explanation figure of Cosine similarity.

Stepwise:
Step 1: Preprocessing the dataset.
Step 2: Creating tags for the recommendation. For creating tags, we merged movie overview, movie genre, movie keywords, movie cast, and movie crew. On combining all those data, we created tags.
Step 3: We applied the stemming process for the filtering. Stemming is the process of stripping a word down to its root or suffix- and prefix-attached word stem. In contrast, a stemming algorithm normalizes language by condensing various word forms to their standard form.The stemming procedure was carried out using "PorterStemmer" from theNLTK package.
Step 4: After stemming filtering, we imported **“CountVectorizer”** from sklearn.feature_extraction.text. It will create a sparse matrix of counts that indicates the frequency of words or concepts in a document.
Step 5: We imported Cosine-similarity from sklearn.metrics.pairwise in order to compare the similarities between various movies. It will determine how connected the different films are. It will plot every movie into a three-dimensional vector and then determine the Cosine distance between various variables.
Step 6: For advice, we developed a function. The recommendation system will then display films that are related.
For collaborative filtering, We will join the datasets "tmdb_5000_movies.csv" and "User ratings.csv"..We will leverage the dataset's userId, movieId, and rating columns for collaborative filtering. For collaborative fi ltering, we will use the “**Matrix Factorization”** approach. By multiplying two different types of entities, matrix factorization can produce latent characteristics. Matrix factorization is used in collaborative filtering to determine the connection between items and user entities. The relationship between people and movie matrices, known as latent features, is used to determine similarity and forecast outcomes based on both user and item entities.

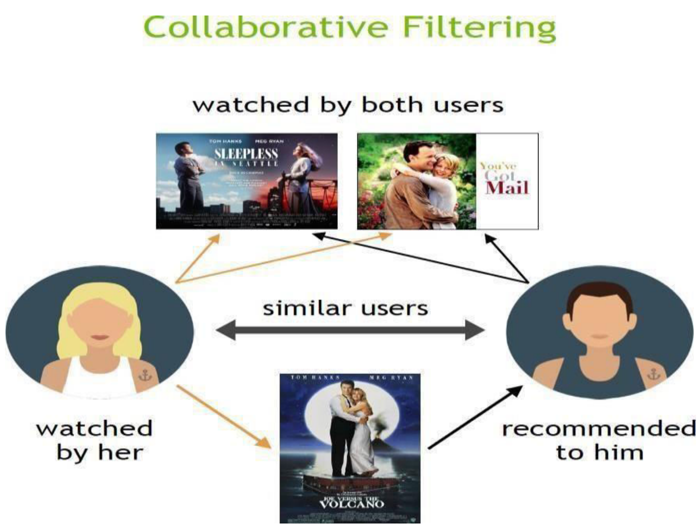

Fig 4:- A explanation picture how a collaborative filtering recommender works.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 4, April 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1204226 |

We will use a feature known as the **"Pearson Correlated Coefficient"** to determine the link between two items. We can determine the link between two quantities using Pearson's correlation coefficient. It provides you with a measurement of how strongly two variables are associated. The Pearson's Correlation Coefficient's value can range from -1 to +1. 1 denotes a strong link between them, while 0 denotes no association.

Pearson Correlation Coefficient

$$r = \frac{n(\Sigma xy) - (\Sigma x)(\Sigma y)}{\sqrt{[n\Sigma x^2 - (\Sigma x)^2][n\Sigma y^2 - (\Sigma y)^2]}}$$

Fig 5: The mathematical equation for Pearson's correlation coefficient.

Stepwise:
Step 1: Preprocessing the data.
Step 2: Merging the ratings and the movies column in a single data frame.\
Step 3: userRatings = ratings.pivot_table(index=['userId'],columns=['title'],values='rating')
Step 4: Then using Pearson's correlation coefficient on user ratings.
Step 5: Written a get similar function for the recommendation.
Step 6: romantic_lover = [("(500) Days of Summer (2009)",5),("Alice in Wonderland (2010)",3),("Aliens (1986)",1),("2001: A Space Odyssey (1968)",2)]. As we can see we passed the movie's name and ratings for the recommendation.
Step 7: For recommendation we called the function.
We also implemented another approach for Collaborative filtering using the "KNN" algorithm. The k-nearest neighbours algorithm, sometimes referred to as KNN or k-NN, is a supervised learning classifier that employs proximity to producing classifications or predictions about the grouping of a single data point.
Both regression and classification problems can be accomplished using the closest neighbour technique. In regression, the goal is to predict a continuous value, like the cost of a cabin; in classification, the goal is to select a label from a limited number of options, like sick or healthy. Apart from KNN we also used the compressed sparse row matrix for creating a sparse matrix in a compressed sparse row format

Stepwise:
Step 1: Preprocessing the data.
Step 2: Merging the ratings and the movies column in a single data frame.
Step 3: movie_pivot=df.pivot_table(columns='userId',index='title',values='rating')
Step 4: Import the compressed sparse row from scipy.sparse and NearestNeighbors from scikit-learn.neighbors.
Step 5: distances,suggestions=model.kneighbors(movie_pivot.iloc[540,:].values.reshape(1,-1)) distances
Step 6: For the recommendation, we have written a function called reco to give the recommendations.
Step 7: We called the function for the recommendation.

IV. CONCLUSION

In order to provide a more precise suggestion based on the fundamentals of user inputs, this project explains many types of movie recommendation systems currently in use. In this project, we covered collaborative filtering. The many uses, benefits, and drawbacks of various types of movie recommendation systems are also covered. The recommendation engine examines the user's prior viewing habits before using this data to look for films that are comparable. The user is then given movie recommendations by the recommendation system. Any recommender system's ultimate goal is to create a model so that the user can make an informed decision.

REFERENCES

1. Movie Recommendation System Jose Immanuvel. J , Sheelavathi. A , Priyadharshan. M , Vignesh. S , Elango. K, International Journal for Research in Applied Science & Engineering Technology (IJRASET), June 2022.
2. Mahesh Goyani and Neha Chaurasiya, "A Review of Movie Recommendation System: Limitations, Survey

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 4, April 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1204226 |

and Challenges", Electronic Letters on Computer Vision and Image Analysis 19(3):18-37, 2020

3. S. K. Raghuwanshi and R. K. Pateriya, "Collaborative Filtering Techniques in Recommendation Systems," in
4. Data, Engineering, and Applications (Springer, Singapore, 2019).
5. R. Ahuja, A. Solanki, and A. Nayyar, "Movie recommender system using k-means clustering and k-nearest neighbor" in Proceeding of the 9th International Conference on Cloud Computing, Data Science and Engineering, (Confluence, 2019), pp. 263–268.
6. Wang, A. P. de Vries, and M. J. T. Reinders, "Unified relevance models for rating prediction in collaborative filtering," ACM Transactions on Information Systems, vol. 26, no. 3, pp. 1–42, June 2008.
7. Paul Marx- Providing Actionable Recommendations: A Movie Recommendation Algorithm with Explanatory Capability, Joseph EulVerlag,2013.
8. C. M. Wu, D. Garg, and U. Bhandary, "Movie Recommendation System Using Collaborative Filtering", IEEE 9th International Conference on Software Engineering and Service Science, 11-15, 2018.
9. C. C. Aggarwal, Recommender Systems The Textbook (IBM T.J. Watson Research Center, 2016).
10. L. T. Ponnam, S. Deepak Punyasamudram, S. N. Nallagulla, and S. Yellamati, "Movie recommender system using item-based collaborative filtering technique," in 1st International Conference on Emerging Trends in Engineering Technology and Science, ICETETS 2016 - Proceedings, (ICETETS, 2016).
11. P. N. V. Kumar and V. R. Reddy, "A Survey on Recommender Systems (RSS) and Its Applications", International Journal of Innovative Research in Computer and Communication Engineering 2(8):5254 - 5260, 2014.