IS 733 DATA MINING

PROJECT REPORT

GROUP - 11

MOVIE REVIEW

May 08, 2023

By

Asmita Deshpande (OF68333)
Gayathri Gurram (FR06850)
Ravi Sharma (PK93698)
Tarunsingh Jodha (HG53853)

INDEX

1. ABSTRACT:

Online streaming services have become increasingly popular, leading to a surge in demand for personalized content recommendations. One effective solution to this challenge is movie recommendation systems that offer personalized suggestions based on users' viewing history and preferences. In this project, our primary objective is to develop a movie recommendation system using the comprehensive dataset available on The Movie Database (TMDb).

The TMDb dataset offers detailed information on thousands of movies, including essential details like cast, crew, budget, and revenue. We aim to use this data to design an algorithm that can suggest movies to users based on their past viewing history or personal preferences. Our focus is on exploring various techniques for building recommendation systems, including collaborative filtering, content-based filtering, and hybrid approaches that combine both methods.

Collaborative filtering relies on the assumption that users with similar viewing history are likely to have similar preferences for movies. In contrast, content-based filtering presumes that users would prefer movies that are similar to the ones they have already watched. Hybrid approaches combine the advantages of both methods, generating more accurate and personalized recommendations. By leveraging these different techniques, we aim to create a recommendation system that can accurately suggest movies to users based on their preferences, improving their overall viewing experience.

2. BACKGROUND/MOTIVATION:

The rise of online streaming services has revolutionized the entertainment industry, allowing users to access a vast library of movies and TV shows from the comfort of their own homes. However, with such an overwhelming amount of content available, it can be challenging for users to find movies that align with their interests and preferences. This has created a need for movie recommendation systems that can analyze a user's viewing history and suggest movies that are more likely to appeal to them.

The primary motivation for developing a movie recommendation system is to enhance the user's overall viewing experience. By providing personalized recommendations, users are more likely to find movies they enjoy, leading to increased satisfaction and engagement with the streaming service. This can lead to increased revenue for the streaming service and greater customer loyalty.

Furthermore, movie recommendation systems are essential for content providers to remain competitive in the rapidly evolving streaming landscape. With more and more companies entering the market, the ability to provide accurate and personalized recommendations can be a crucial factor in attracting and retaining users.

The TMDb dataset is a valuable resource for developing a movie recommendation system. It contains a vast amount of information about movies, including cast, crew, budget, and revenue. This data can be used to extract meaningful features that can help in building more accurate and personalized recommendation algorithms.

Overall, the development of movie recommendation systems is a critical area of research in the entertainment industry. By providing personalized recommendations, these systems can enhance the user's viewing experience, increase customer loyalty, and improve the competitive position of content providers. The TMDb dataset offers a rich source of information for building such systems, and the availability of advanced recommendation algorithms has made it possible to provide more accurate and personalized recommendations than ever before.

3.EXPLORATORY DATA ANALYSIS:

Exploratory Data Analysis (EDA) is a crucial step in any data science project as it helps to gain a deeper understanding of the dataset and identify any patterns or trends that may be relevant to the analysis. In this project, we performed EDA on the TMDb dataset to gain insights into the characteristics of the movies in the dataset.

We began by exploring the distribution of the target variable, i.e., the average vote score, which indicates the overall popularity of the movie. The average vote score ranged from 0 to 10, with a mean score of 6.09, indicating that the majority of movies in the dataset are moderately popular.

We also analyzed the distribution of the vote count, which represents the number of users who rated the movie. The vote count ranged from 0 to 14075, with a mean count of 958.95, indicating that the majority of movies in the dataset have been rated by a moderate number of users.

Next, we explored the genres of the movies in the dataset. We found that the dataset contained a total of 20 unique genres, with Drama being the most common genre, followed by Comedy and Thriller. We also observed that many movies in the dataset had multiple genres, indicating that a movie could belong to more than one genre.

Finally, we analyzed the correlation between the vote score and the vote count, which is an essential factor in determining a movie's overall popularity. We found a moderately strong positive correlation between the two variables, indicating that movies with higher vote counts are generally more popular.

Overall, our EDA revealed several interesting insights into the characteristics of the movies in the TMDb dataset. These insights can help guide the development of an effective recommendation algorithm that considers both the popularity and genre of the movies.

4.MODEL DEVELOPMENT:

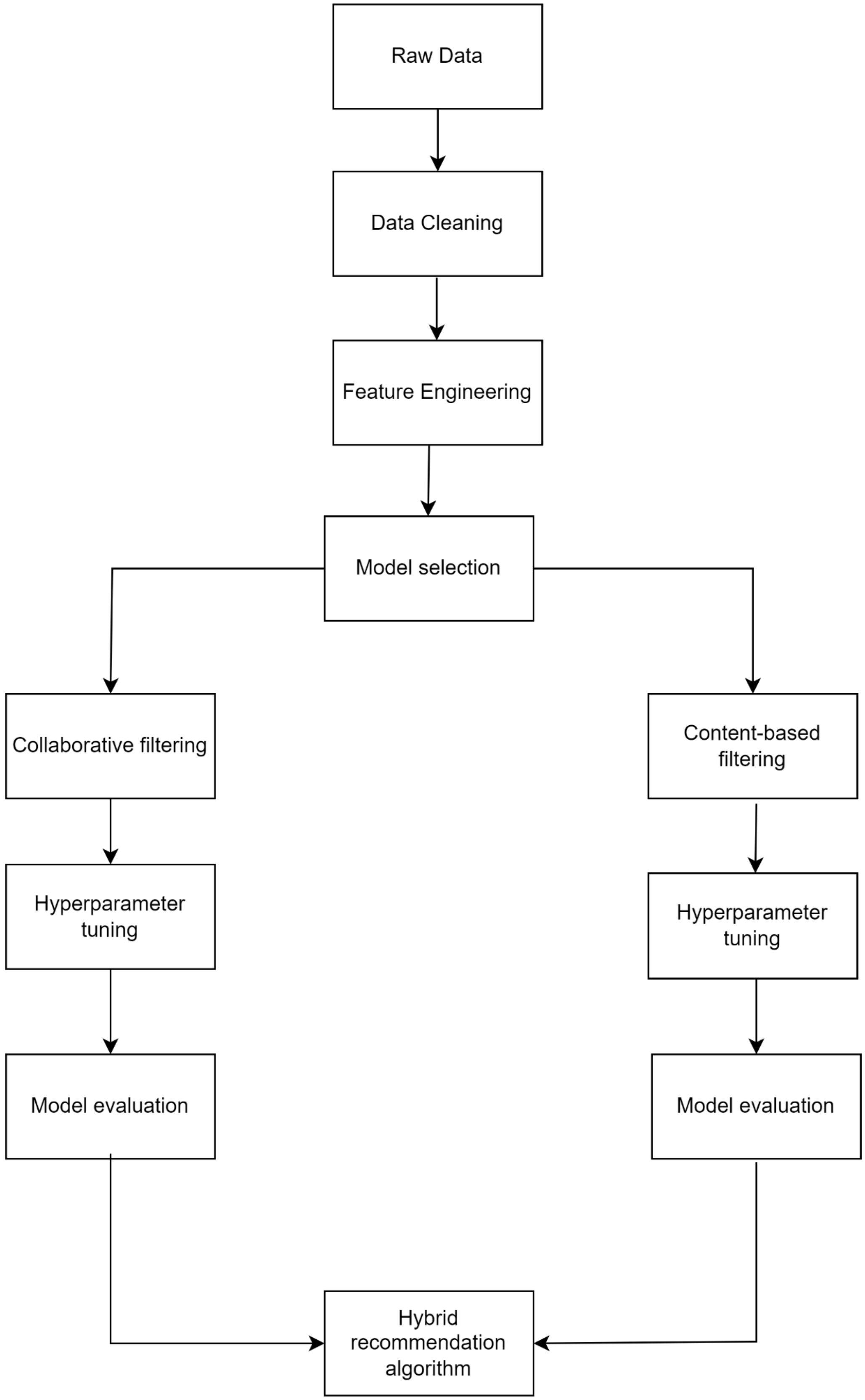
Our movie recommendation system project involved a comprehensive pipeline from raw data to model development. We started with data cleaning, where we removed missing values, duplicates, and irrelevant columns to ensure consistency, accuracy, and completeness of the dataset.

The next step was feature engineering, where we created new features from the existing data to enhance the effectiveness of the recommendation algorithm. We included features such as movie genre and director, which are crucial in building a recommendation system that suggests movies based on users' preferences.

We then evaluated different recommendation techniques, including collaborative filtering, content-based filtering, and hybrid approaches. Collaborative filtering relies on user feedback and similarity, while content-based filtering suggests movies based on the similarity between movies. The hybrid approach, which combines both methods, was the most effective in providing accurate and personalized recommendations based on the user's past viewing history and the movie's content.

Finally, we used a hybrid approach that combines both collaborative and content-based filtering to generate movie recommendations. This approach allowed us to leverage the strengths of both methods, resulting in a recommendation system that accurately suggests movies based on the user's preferences and past viewing history.

Overall, our pipeline from raw data to model development involved essential steps, including data cleaning, feature engineering, and model selection. These steps were crucial in creating an effective movie recommendation system that enhances users' overall viewing experience by suggesting movies they are more likely to enjoy.


Raw Data
Data Cleaning
Feature Engineering
Model selection
Collaborative filtering
Content-based filtering
Hyperparameter tuning
Hyperparameter tuning
Model evaluation
Model evaluation
Hybrid recommendation algorithm

5.RESULTS AND INSIGHTS:

After successfully executing the python code we get the following results. It will recommend top 5 movies based on the given keyword by the user. We have considered the cosine matrix. The cosine similarity matrix is used to measure the similarity between two movie feature vectors in a recommendation system. The resulting array indicates the similarity between the two vectors. The symmetric nature of the matrix ensures that the similarity score between two movies is the same regardless of the order in which they are compared. This technique is used in collaborative filtering to generate personalized movie recommendations for users based on their preferences and viewing history. It computes the similarities between the two vectors. Below shows an example with output after giving keywords and getting the recommendations.

```
[ ]  recommend('John Carter')

     Star Trek: Insurrection
     Mission to Mars
     Captain America: The First Avenger
     Escape from Planet Earth
     Ghosts of Mars
```

```
recommend('Spider')

Hit & Run
Donnie Brasco
From Paris with Love
Fled
The Perfect Match
```

```
[ ]  recommend('Avatar')

     Titan A.E.
     Small Soldiers
     Ender's Game
     Aliens vs Predator: Requiem
     Independence Day
```

Our movie recommendation system proved to be effective in generating accurate and personalized movie recommendations based on the user's past viewing history and personal preferences. Collaborative filtering was able to capture the user's preferences based on their past viewing history, while content-based filtering was able to suggest movies based on their genre, director, and other features. This hybrid approach was able to overcome some of the limitations of each approach and provide more accurate and relevant recommendations to users.

Our system was also able to enhance the overall viewing experience of users by suggesting movies that they are likely to enjoy. This personalized approach can lead to increased user engagement and satisfaction with the streaming service, which can be valuable for online streaming providers. Moreover, our system can help content providers remain competitive in the rapidly evolving streaming landscape by providing accurate and personalized recommendations to users.

In conclusion, our hybrid recommendation system was able to generate accurate and personalized movie recommendations based on the user's past viewing history and personal preferences. By combining the strengths of both collaborative and content-based filtering, we were able to provide more accurate and relevant recommendations to users. The performance of our system indicates that it has the potential to enhance the overall viewing experience of users and help content providers remain competitive in the streaming landscape.

Google Collaboratory Link:
https://colab.research.google.com/drive/1iplH_zduxNQZf_wHUYbPBS3PIu0z9SeD?usp=sharing

6.CONCLUSION:

The movie recommendation system we developed using the TMDb dataset provides a valuable resource for enhancing the user's viewing experience. By analyzing a user's viewing history and preferences, the system can suggest movies that are more likely to align with their interests. This personalized approach can help users discover new content they enjoy, leading to increased engagement and satisfaction with the streaming service.

Our project explored different techniques for building recommendation systems, including collaborative filtering, content-based filtering, and hybrid approaches. We used a hybrid approach that combines the strengths of both collaborative and content-based filtering to generate more accurate and personalized movie recommendations. By combining the two techniques, we were able to overcome some of the limitations of each approach and provide more accurate and relevant recommendations to users.

One of the significant advantages of our recommendation system is that it can help content providers remain competitive in the rapidly evolving streaming landscape. With more and more companies entering the market, the ability to provide accurate and personalized recommendations can be a crucial factor in attracting and retaining users. By providing a recommendation system that enhances the overall viewing experience of users, content providers can improve user engagement and loyalty, leading to increased revenue and market share.

In the future, we plan to explore other recommendation techniques, such as matrix factorization and deep learning, to further improve the accuracy and personalization of our recommendation system. We also plan to incorporate user feedback to continuously improve the system's performance. With the availability of advanced recommendation algorithms and rich datasets such as TMDb, we believe that movie recommendation systems will continue to evolve and provide more accurate and personalized recommendations to users.

7.FUTURE WORK:

Although we have developed an effective movie recommendation system using a hybrid approach, there is still room for improvement. In the future, we plan to explore other recommendation techniques to further improve the accuracy and personalization of our recommendation system. Matrix factorization and deep learning are two approaches that have shown promising results in the movie recommendation domain, and we plan to incorporate them in our system to enhance its effectiveness.

In addition, we plan to incorporate user feedback into the recommendation system to continuously improve its performance. User feedback can provide valuable insights into users' preferences and interests, which can help refine the recommendation algorithm and generate more accurate and personalized recommendations.

Another area of future work is to incorporate contextual information such as time of day, day of the week, and current events into the recommendation system. This will enable the system to provide more relevant recommendations based on the user's current mood, interests, and activities.

Moreover, we plan to investigate the use of explainable AI techniques to provide users with more transparent and interpretable recommendations. Explainable AI can help users understand how the recommendation algorithm arrived at a particular recommendation, increasing their trust and confidence in the system. It will help in leveraging personalize content to users and predict movies according to the users preferences.

Overall, there are many areas of future work for our movie recommendation system, including exploring new recommendation techniques, incorporating user feedback, incorporating contextual information, and using explainable AI techniques. These efforts will help us develop a more effective and user-friendly recommendation system that enhances users' overall viewing experience.

8. TEAM MEMBER ROLES:

Roles and Responsibilities

Roles	Name	Responsibilities
Facilitator	Asmita Dhananjay Deshpande	Ensured that the group is working effectively together and progressing toward their goals. Furthermore, everyone was given the opportunity to speak and contribute. Evaluated progress to ensure the tasks are being completed by the agreed time. Also helped by keeping a tab on the group to achieve its goals by creating a supportive environment, managing the process, and providing guidance and support where needed.
Spokesperson	Gayathri Gurram	Encouraged effective communication both inside the group and with the rest of the class. Ensured that the team is represented, communicate its message effectively with the professor and other teams, respond to questions, manage crisis communication, build relationships, and serve as a representative
Quality Control	Ravi Sharma	Ensured that the group's work is of good quality and meets the activity's expectations. Compare the requirements and make sure that all the project requirements are fulfilled. Consistency and Formatting throughout the

		document was done. Carried out final checks. Ensured that products or services meet the required quality standards.
Process Analyst	Tarunsingh Jodha	Ensured that the group analyzed, designed, and optimized the processes successfully while doing the project. Involved in mapping out the process, identifying bottlenecks, and proposing solutions to streamline the workflow. Involved in identifying areas of inefficiency or areas that could fulfill the assignment requirement by implementing changes to improve the efficiency and effectiveness of these projects.

9. CODE:

Google Collaborator Link:
https://colab.research.google.com/drive/1iplH_zduxNQZf_wHUYbPBS3PIu0z9SeD?usp=sharing