# Introduction

Anime Recommendation System is a machine-learning model designed to provide personalized anime recommendations to users. With the vast number of anime titles available today, it can be challenging for users to navigate the sea of choices and find content that matches their interests. This is where recommendation systems come in.

The anime recommendation system uses machine learning algorithms to analyze the user's watch history and generate a list of anime that the user is likely to enjoy. By considering factors such as the user's past viewing preferences, ratings, and reviews, the system can suggest anime that align with the user's tastes and preferences.

To develop a highly accurate and effective recommendation system, the model is trained using a large dataset of over +12,000 anime titles and over 7.8 million user reviews. By analyzing this vast dataset, the model can identify patterns and similarities between different anime titles and user preferences.

Through this process, the system can suggest anime titles that are not only similar to the user's previous viewing history but also consider other relevant factors such as genre, release date, and popularity. This helps users discover new anime titles that they may not have found otherwise and enhances their overall anime viewing experience.

Overall, the anime recommendation system is an innovative and sophisticated tool that can provide anime fans with a more personalized and enjoyable viewing experience by suggesting titles that are tailored to their unique preferences and interests.

# Background

Anime recommendation systems have become increasingly popular in recent years, as more and more people around the world are embracing anime as a form of entertainment. With the rise of streaming platforms like Netflix, Hulu, and Crunchyroll, anime has become more accessible to a wider audience, leading to an explosion in popularity.

However, the sheer number of anime titles available can be overwhelming for many viewers, especially those who are new to the genre. This is where recommendation systems come in, helping users navigate the vast array of choices and find content that matches their interests.

The development of anime recommendation systems is a complex and challenging task, as it requires the analysis of vast amounts of data and the use of sophisticated machine-learning algorithms. The goal is to provide personalized recommendations to each user, based on their viewing history, ratings, and reviews.

To achieve this, researchers have employed a variety of techniques, including collaborative filtering, content-based filtering, and hybrid models. Collaborative filtering involves analyzing the viewing patterns of multiple users to identify commonalities and make recommendations based on those similarities. Content-based filtering, on the other hand, focuses on analyzing the attributes of individual anime titles, such as genre, plot, and characters, and making recommendations based on those factors. Hybrid models combine both approaches to provide more accurate and effective recommendations.

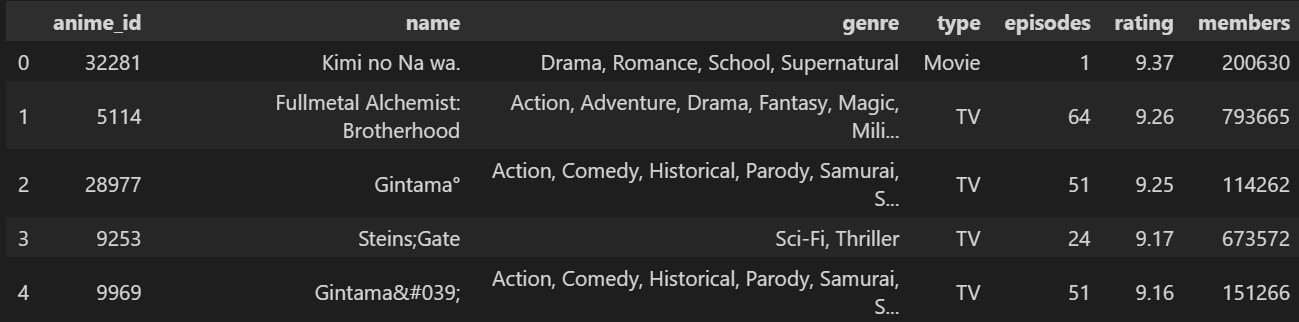
As the popularity of anime continues to grow, the development of more sophisticated and effective recommendation systems will be crucial in helping users find content that matches their interests and preferences. This will not only enhance the viewing experience for users but also help anime producers and streaming platforms better understand their audiences and develop more targeted content.

# Data Collection

The success of an anime recommendation system is heavily reliant on the quality and quantity of data used to train the machine learning algorithms. Therefore, data collection and preparation are essential steps in building an effective recommendation system.

To collect data for the anime recommendation system, we have collected [anime databases](https://www.kaggle.com/datasets/CooperUnion/anime-recommendations-database) from Kaggle to help support the case. The anime recommendation database has a long list of anime with a large community of user votes, making it an ideal source of data for building a recommendation system.

After downloading the dataset, the dataset can be loaded with the Python notebook, where we can use df.head() to obtain the list of the first 5 rows of results.



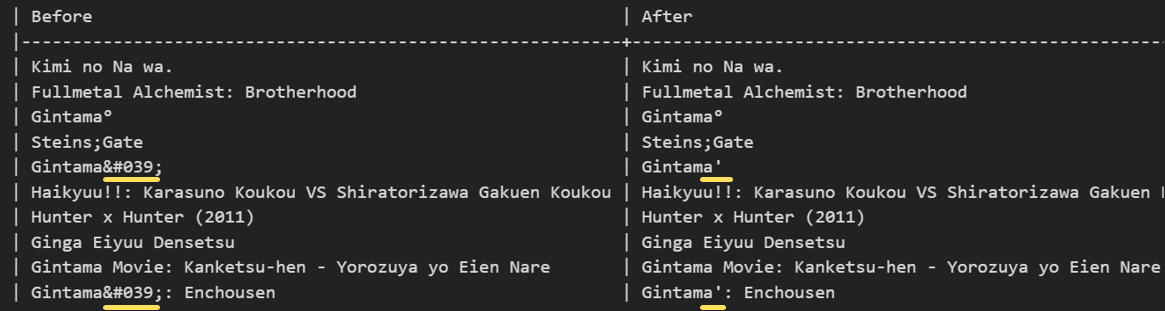
# Data Preprocessing

## Decoding HTML Characters

In most datasets, it's common to encounter HTML characters, which are used to encode special characters or symbols that aren't available in standard character sets. These characters can include symbols like &, <, and >, as well as special characters like apostrophes and quotation marks.

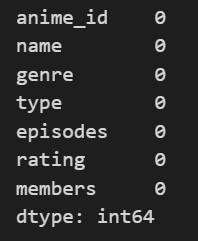
However, when working with natural language processing (NLP) tasks like text classification or sentiment analysis, it's important to remove these HTML characters from the dataset, as they can interfere with the accuracy of the machine learning model. For example, if an apostrophe is encoded as ', the machine learning model may not recognize it as a valid character and could misinterpret the meaning of the text.

To remove these HTML characters, a process called HTML entity decoding is used. This involves replacing the HTML entities with their corresponding characters. For example, & would be replaced with &, ' would be replaced with ', and so on.



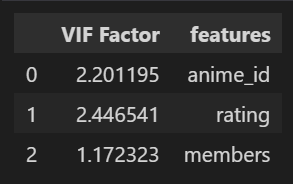
## Filtering Null/Unknown Values

To ensure the accuracy of the anime recommendation system, any "Unknown" values in the dataset are first replaced with null values. The dataset is then filtered to remove any null or N/A values using the df.dropna() function. Additionally, any anime titles with less than 100 user reviews are also filtered from the dataset.

## Checking Multi-Collinearity

To check for multi-collinearity within the anime recommendation system data, we used the Variance Inflation Factor (VIF). The results of the VIF analysis showed that all values were below 5, indicating that there was no evidence of multi-collinearity within the dataset.

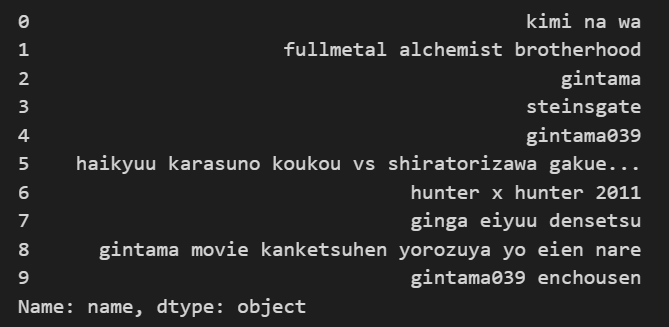


# Exploratory Data Analysis (EDA)

To perform exploratory data analysis (EDA) on the anime recommendation system, we will use a variety of data visualization techniques to gain insights into the relationships and patterns within the data. This will involve generating graphs, charts, and other visual representations of the data to help us identify any trends or anomalies. By performing EDA, we can gain a better understanding of the dataset and potentially uncover new insights that can help improve the accuracy of the recommendation system.

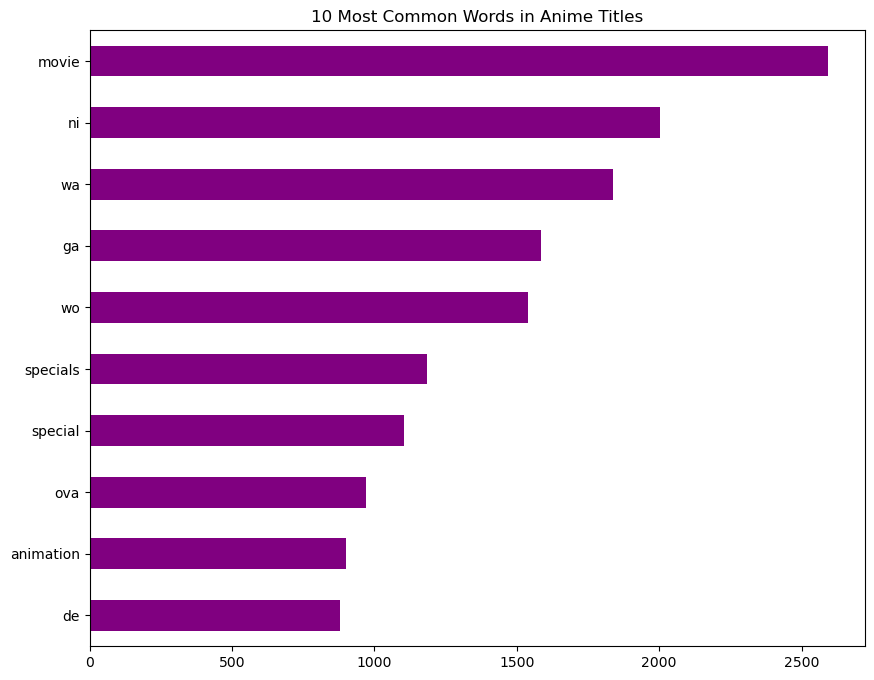
## Removing Punctuations

By removing punctuation from the anime dataset, we can create a search method that is more efficient and accurate. The addition of a cosine similarity method further improves accuracy, allowing users to find similar animes based on typed queries. This combination of techniques helps to ensure that the recommendation system provides highly relevant results to users.



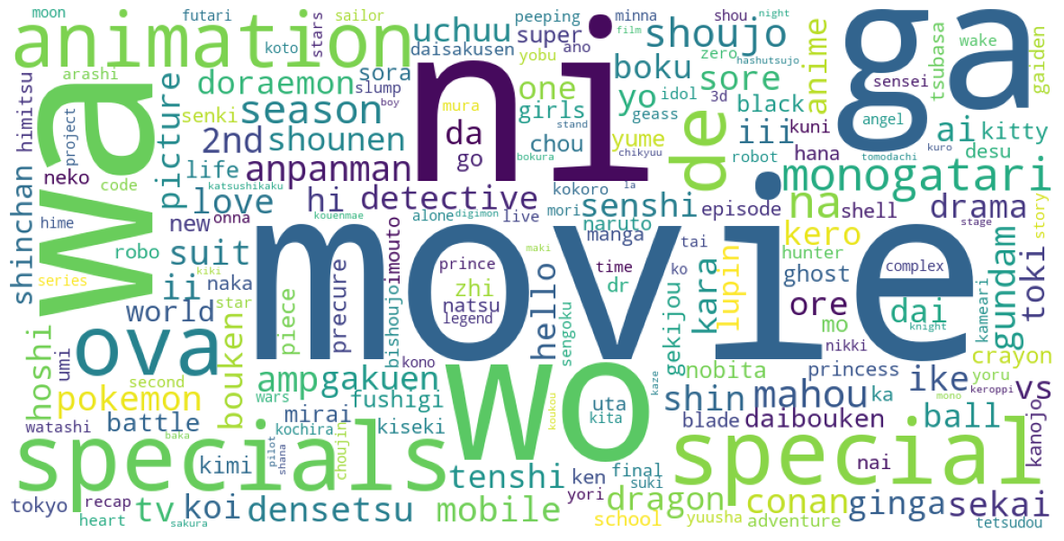
## Word Count

Word count analysis of the anime dataset revealed that the term "movie" appeared more than 2,500 times. To improve the accuracy of the search results, we implemented TF-IDF to remove external references like movies. By doing so, we were able to focus on more relevant terms and produce more accurate search results for users.



## Word Cloud

A word cloud analysis of the anime dataset revealed that certain terms like "special", "specials", "animation", and "season" were highly repetitive. However, these terms should not be factored into the recommendation system as they are not directly related to the anime itself. Specials, for example, are typically watched after the anime series is completed. By identifying these irrelevant terms, we can improve the accuracy of the recommendation system and provide more relevant results to users.

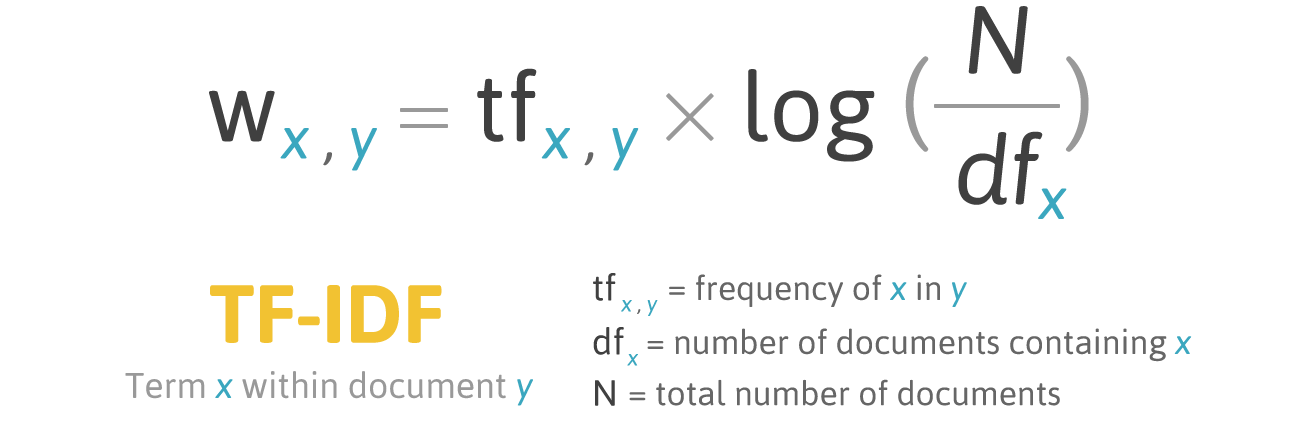


# Feature Engineering

We plan to use feature engineering techniques to improve the accuracy of the anime recommendation system. This will involve optimizing the TF-IDF vectorizer to identify and weight important terms in the dataset. We will also apply feature engineering to the TF-IDF based word cloud to remove irrelevant terms. Finally, we will use cosine similarity to refine the search algorithm, improving the accuracy of the recommendation system overall.

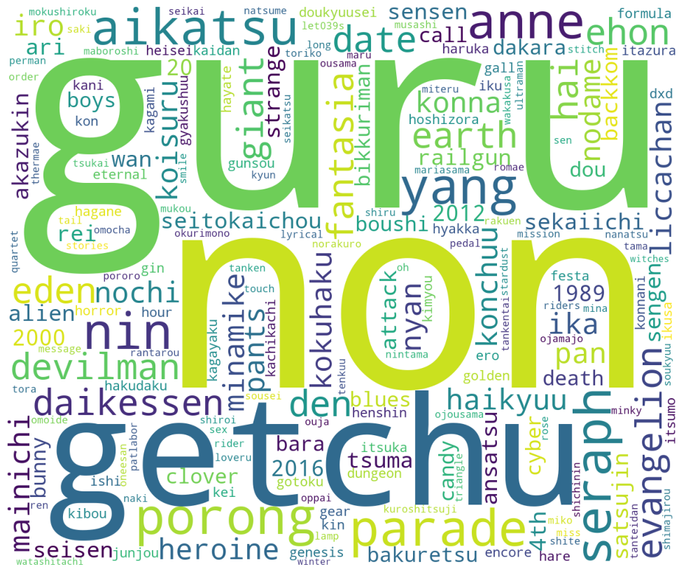
## TF-IDF Vectorizer

TF-IDF is a method used to assess the importance of words within a document. It does this by analyzing the frequency of a word in a document and comparing it to the frequency of the word across all the documents in the dataset. By taking into account both the frequency and relative rarity of a word, TF-IDF can help to identify the most important terms in a dataset.



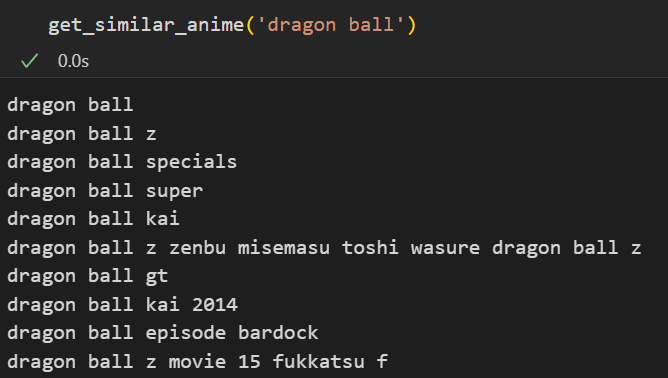
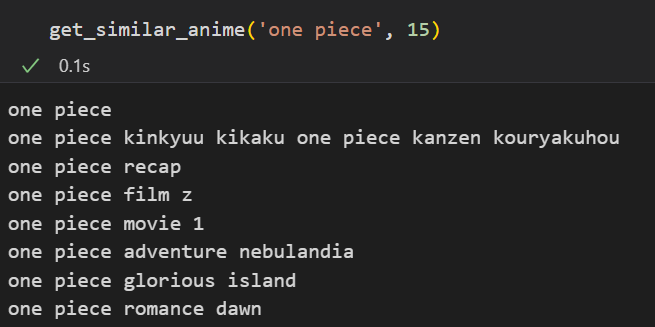
## TF-IDF-Based Word Cloud

TF-IDF can help to improve the accuracy of the anime recommendation system by identifying and weighting important terms in the dataset. By removing irrelevant terms like "animation" and "specials", the search algorithm can more accurately identify the most relevant recommendations for a user based on their search query. In this way, TF-IDF can help to refine the search algorithm and provide more accurate recommendations overall.



## Cosine Similarity

Based on the TF-IDF word list, we can create a pairwise similarity score for each anime in the dataset. If the word that the user is searching for is too similar to a word in the TF-IDF word list, the algorithm will return the entire anime name as the result. By using this approach, we can create a list of the most relevant anime recommendations based on the user's search query, improving the overall accuracy of the recommendation system.

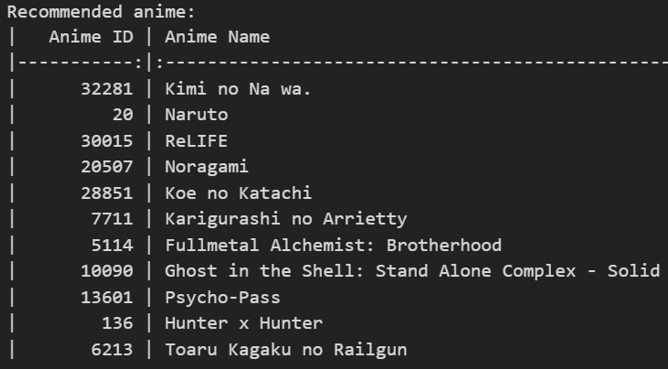
# Model Selection and Training

For the anime recommendation problem, we considered three different models: content-based filtering, collaborative filtering, and a hybrid model that combines both approaches.

Content-based filtering was chosen due to its ability to recommend anime based on similarities in the features of the anime, such as genre, theme, and plot. Collaborative filtering was chosen for its ability to recommend anime based on the user's watch history and the preferences of similar users. The hybrid model was chosen to leverage the strengths of both models and provide a more accurate and personalized recommendation.

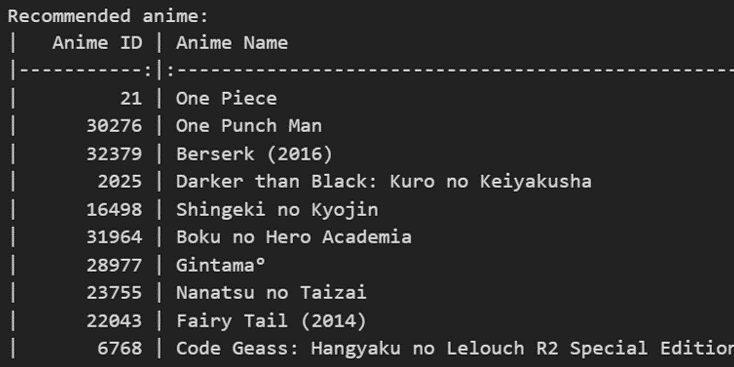
## Content-Based Filtering

Content-based filtering is a recommendation technique that recommends items to users based on similarities in their content or attributes. In the case of anime, the system would recommend anime with similar genres, ratings, and liked by the number of people of the said anime that the user has previously watched and liked. Content-based filtering is effective when there is a significant amount of data available on the content attributes of the items being recommended.

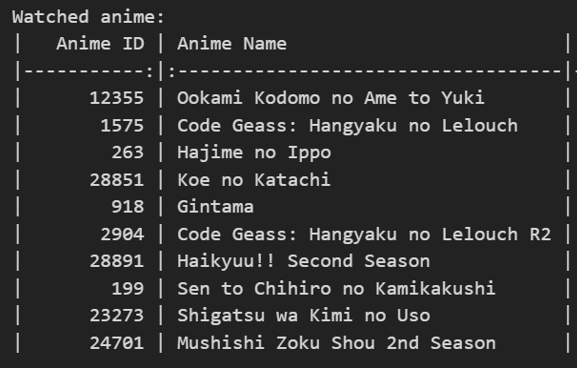
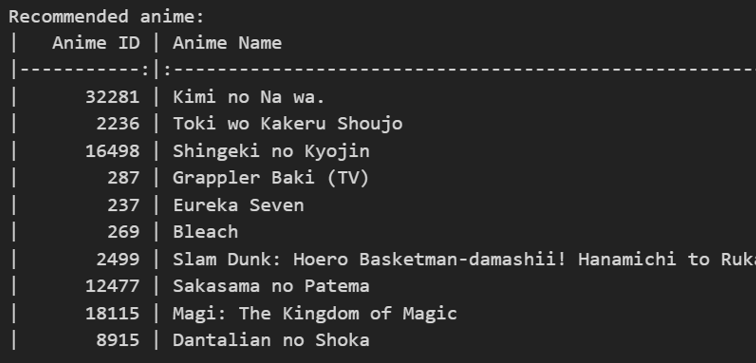
## Collaborative Filtering (CF)

Collaborative filtering is a popular recommendation technique that recommends items based on the preferences and behaviors of similar users. In the case of anime, the system would recommend anime to a user based on the ratings and watch history of other users who have similar preferences to that user. CF is effective when there is a large amount of user data available, and when users have similar preferences. However, it can also suffer from the cold-start problem, where users with less watched anime list marks it as insufficient data for accurate recommendations.

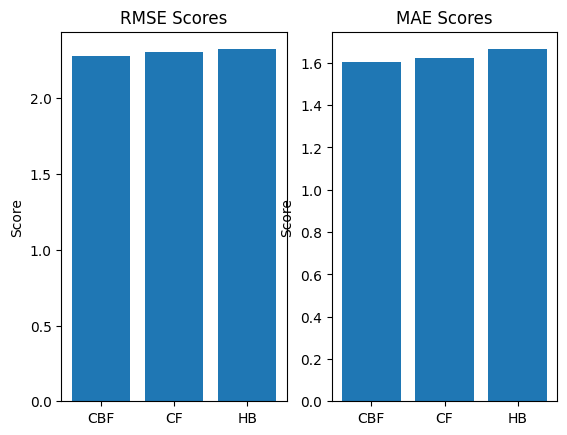
## Hybrid Model

A hybrid model combines two or more recommendation techniques to improve the accuracy of the recommendations. In the case of anime recommendation, a hybrid model is combining both content-based filtering and collaborative filtering to take advantage of the strengths of both methods. For example, the content-based approach is being used to recommend anime based on similarities in their genres, while the collaborative filtering approach is being used to recommend anime based on the preferences of similar users. By combining these techniques, a hybrid model can provide more accurate and personalized recommendations to users.

# Model Evaluation and Results

To evaluate the performance of our anime recommendation system, we used the root mean square error (RMSE) and mean absolute error (MAE) metrics on the test data. Our Content-Based Filtering model achieved a mean RMSE of 2.28 and a mean MAE of 1.60. The Collaborative Filtering model had a mean RMSE of 2.31 and a mean MAE of 1.62. Finally, our Hybrid model had a mean RMSE of 2.32 and a mean MAE of 1.66.



Overall, our models had relatively similar performance in terms of RMSE and MAE. However, the Content-Based Filtering model had the lowest RMSE and MAE scores, indicating that it is the most accurate model for our recommendation system.