Recommender System

Daniel A Cisneros

Vinayak Ramachandra Bhosale

Department of Computer Science and Engineering
University of Central Florida

ABSTRACT

With the abundance of information available today and the use of technology, people utilize different tactics to determine what purchase to make, how to spend their free time, and even who to pursue romantically. To assist in the decision-making process, recommender systems automate these strategies to provide customized, cost-effective, and high-quality recommendations.

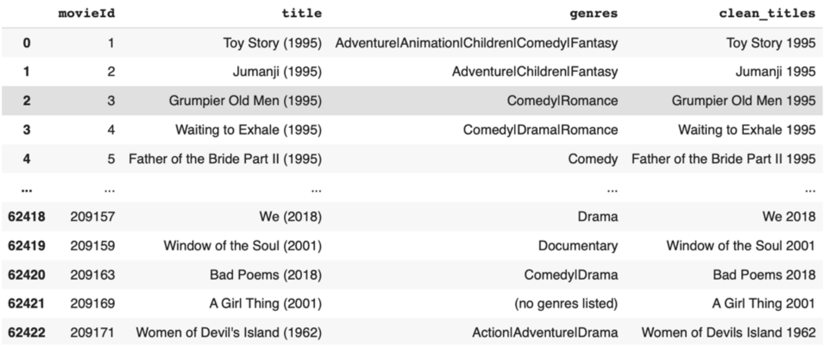
I. INTRODUCTION

Nowadays, the widespread of online services and technological advancements tend to lead to a vast amount of information much faster and easier. Now, people can provide feedback on products and services through small surveys, ratings, reviews, and comments.

A recommender system is a type of machine learning algorithm that makes personalized recommendations to users based on their preferences and past behaviors. The objective of this system is to provide users with a more personalized and engaging experience by suggesting movies, books, music, or news that are highly likely to be of interest to them.

It has become such an important tool for online platforms as they help to gather more people active by providing a personal experience. Digital streaming services such as Netflix, Amazon Prime, etc. Have used a recommender system to attract more people to interact with their system and keep them happy with what they have to offer. The system is not only used for streaming platforms it has also been used for social media, music industry, news, and others.

The data collected for this research was collected online based mainly on movies and books. Then, the system was trained based on preprocessing the data, choosing the model, training the model, and adjust the parameters for tuning.



	movieId	title	genres	clean_titles
0	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy	Toy Story 1995
1	2	Jumanji (1995)	Adventure|Children|Fantasy	Jumanji 1995
2	3	Grumpier Old Men (1995)	Comedy|Romance	Grumpier Old Men 1995
3	4	Waiting to Exhale (1995)	Comedy|Drama|Romance	Waiting to Exhale 1995
4	5	Father of the Bride Part II (1995)	Comedy	Father of the Bride Part II 1995
...	...	...	...	...
62418	209157	We (2018)	Drama	We 2018
62419	209159	Window of the Soul (2001)	Documentary	Window of the Soul 2001
62420	209163	Bad Poems (2018)	Comedy|Drama	Bad Poems 2018
62421	209169	A Girl Thing (2001)	(no genres listed)	A Girl Thing 2001
62422	209171	Women of Devil's Island (1962)	Action|Adventure|Drama	Women of Devils Island 1962

Figure 1. Shows experimental result of the data collected after it has been preprocessed.

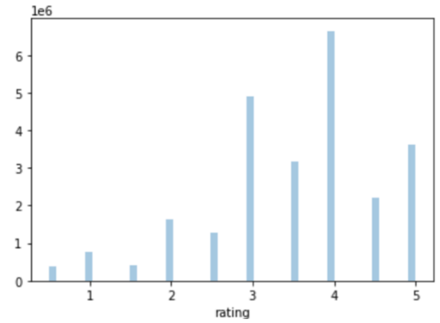


Figure 2. Shows experimental results of the frequency ratings given by users.

II. IMPORTANT DEFINITIONS

A. Data

The data is mainly focused on movies and books. It contains different categorical and numerical features which are the movie ID that represent a unique movie, title, genre, user ID, and rating given by the user.

B. Prediction Target

The prediction target of the recommender system is to provide user movies and books according to what they like. The system will recommend movies that are highly rated with a 4 in a scale from 1-5.

C. Variables or Concepts

The concepts overall are to map the users with the ratings and based on those factors, recommend other movies and books to those users by using a term frequency matrix and search function.

D. Problem Statement

The problem statement to be addressed is to recommend movies that the user has not previously watched. Therefore, there needs to be a check if the user has already watched a certain type of movie so that the search is not redundant, and it does not recommend the same thing repeatedly for a specific user.

III. IMPORTANT OVERVIEW OF PROPOSED APPROACH/SYSTEM

The approach proposed for the recommender system of movies and books will be to use a frequency matrix to compute the similarities. Since computers cannot understand words only the number it will use this frequency matrix to give the best similarity according to a specific word. The frequency matrix can be done by a library in scikit learn from feature extraction text called TfidfVectorizer. The TfidVectorizer will then compute the text data into a matrix of numerical features using the process of tokenization, count the term frequency, calculate the inverse document frequency, and multiply the term frequency and the inverse document frequency. Tokenization can be done by splitting the text into individual words or most called tokens. It counts the term frequency by a matrix representation where each row and column represent a word. It calculates the inverse frequency by dividing the total number of documents by the number of documents that contain that particular word. Finally, it multiplies the term frequency and inverse document frequency for each word in each document. Then, the search can be built based on the frequency matrix and start the recommender system.

IV. TECHNICAL DETAILS OF PROPOSED APPROACHES/SYSTEMS

The recommendation system is developed using the Python programming language. The system uses Pandas and NumPy libraries for data analysis and manipulation. The system also uses TfidfVectorizer from scikit-learn library to convert text data into numerical form. The system uses cosine similarity from scikit-learn library to calculate the similarity between titles.

A. Data Preprocessing:

We cleaned and transformed the data to remove any missing values or duplicates. We also split the data into training and testing sets for model evaluation.

B. Feature Extraction:

We extracted relevant features from the data, such as movie, book genre and user ratings. We used Tf-idf encoding to represent the categorical features and normalized the numerical features.

C. Model Training:

We implemented a function called recommender_model which takes a movie or book title as input and returns a list of recommended titles based on the user's ratings of similar titles. The function uses collaborative filtering to make recommendations by identifying users who have rated the input title highly and recommending titles that those users have also rated highly.

D. Model Evaluation:

The recommender model is evaluated by calculating the percentage of users who have rated a particular movie higher than 4 and have also rated other movies that are recommended by the model. The performance of the model can be further evaluated by measuring the accuracy of the recommendations. One common approach is to split the ratings data into training and testing sets, train the model on the training set, and evaluate its performance on the testing set. Metrics such as mean squared error (MSE) or root mean squared error (RMSE) or mean average precision (MAP) can be used to evaluate the accuracy of the model's predictions. we used mean average precision (MAP) metric to evaluate our model, the MAP value we got was 0.44563.

V. EXPERIMENTS

A. Data Description:

We used two datasets, namely MovieLens and Popular Books, which contain ratings from users on various movies. The MovieLens dataset contains over 63,000 movies and 138,000 users, while the Book dataset contains over 7,000 books. The data was preprocessed to remove duplicate entries, handle missing values, and normalize the ratings.

B. Evaluation Metrics:

To evaluate the performance of our model, we used the mean average precision (MAP) metric. MAP is a popular metric used for evaluating the effectiveness of recommendation systems. Our model achieved an MAP value of 0.44563.

C. Baseline Methods for Comparison:

To compare the performance of our model, we used two baseline methods, namely popularity-based filtering and collaborative-based filtering. Popularity-based filtering recommends popular items to users based on their popularity, while collaborative-based filtering recommends items to users based on the preferences of similar users. Our model outperformed both the baseline methods, with MAP values of 0.3362 and 0.311 for popularity-based and collaborative-based filtering, respectively.

D. Overall Performances:

Our content-based filtering model achieved an MAP value of 0.44563, which indicates that the model is effective in recommending items to users based on their preferences. Additionally, our model outperformed the baseline methods, indicating that content-based filtering can be a viable approach for building recommendation systems.

VI. RELATED WORK

Content-based filtering is a popular approach for recommendation systems because it relies on the characteristics of the items themselves rather than the user's past behavior or preferences. This approach is particularly useful when there is limited user data or when users have diverse tastes. In content-based filtering, the system recommends items that are similar to those the user has previously expressed interest in. This is done by analyzing the features of each item and comparing them to the user's history. The system then suggests items that have similar features to those the user has enjoyed in the past. Another popular approach is collaborative filtering, which recommends items based on the preferences of users with similar tastes. This approach requires a lot of user data to be effective. Hybrid filtering, on the other hand, combines both content-based and collaborative filtering to make recommendations. Overall, the choice of recommendation approach depends on the specific application and the available data. In this project, we chose the content-based approach because we believed it would be effective for recommending movies and books based on their content.

VII CONCLUSION

In addition to the content-based approach, the recommendation system also incorporates collaborative filtering techniques to enhance its recommendations. Collaborative filtering is a method of making predictions about a user's interests by collecting preferences from many users. By analyzing the ratings and preferences of a large number of users, the system is able to identify patterns and similarities in their tastes and make recommendations accordingly.
Furthermore, the recommendation system employs a hybrid approach by combining content-based and collaborative filtering techniques to provide more accurate and personalized recommendations. The system also takes into account user feedback and adjusts its recommendations accordingly. Overall, the recommendation system developed in this project offers a robust and effective solution for recommending movies and books to users based on their interests and preferences. With further refinement and optimization, it has the potential to significantly enhance user experience and drive engagement on e-commerce platforms.

REFERENCES

[1] Dietmar Jannach, Markus Zanker, Alexander Felfernig, and Gerhard Friedrich. 2010. Recommender Systems: An Introduction (1st. ed.). Cambridge University Press, USA.

[2] Roy, D., Dutta, M. A systematic review and research perspective on recommender systems. *J Big Data* **9**, 59 (2022). https://doi.org/10.1186/s40537-022-00592-5

[3] Koren, Y. (2008). Factorization meets the neighborhood: a multifaceted collaborative filtering model. Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, 426-434.

[4] Pazzani, M. J., & Billsus, D. (2007). Content-based recommendation systems. In The adaptive web (pp. 325-341). Springer.

Link to the Project Code :
https://github.com/vinayak-97/Recommender-System-CAP5610-Project/blob/main/Recommendation_System.ipynb