Movie Recommendation System

Amit Dilip Kini, Ashini Anantharaman, Niranjana Sathish Avilery, Praveen Chandrasekaran, Vigneshwaran Ravichandran
DSCI 633 Foundations of Data Science - Fall 2021. Professor: Dr. Nidhi Rastogi, Rochester Institute of Technology

***Abstract*—*Consumers of over-the-top (OTT) video content services bank on the recommendations provided to make a choice for which movie or show to watch next. Content recommendation systems help OTT platforms offer a staunch user experience by reducing the time for content discovery and increasing the time for content consumption. This helps reduce churn and establish brand loyalty. This paper leverages the content based and collaborative recommendation models to build a system to recommend movies using a combination of the top movies in a genre and the users' consumption history. The exploratory data analysis determines that the genre of a movie is the most critical aspect of a user's consumption pattern. The content-based recommendation model uses the genre of a given movie to recommend the top movies of that genre. The collaborative model analyzes the user's rating data and finds users with similar preferences to derive the recommended movies. The paper also explores and opens the concept of hybrid models for recommendations engines of the new age.*

I. Introduction

The OTT entertainment industry is a 129 billion USD business in 2021 and is estimated to be doubled to around 210 billion USD by 2026 [1]. One of the key differentiators to becoming a successful platform, is understanding what the users want to enhance the user experience. With increasing competition between numerous platforms creating and licensing premium high-quality content, the key deciding factor for success is content discovery. A Stat from 2017 show viewers spend 51 minutes per day searching for shows to watch [2] and that hasn't changed much today. This leads to churn with users switching to other platforms even while a platform has high quality content. With the huge amount of data generated during user interactions, researchers thrive on it to create mechanisms that can predict users' next step or help the user in making his decision for the next action he takes. Recommendation engines have been a research topic since the advent of e-commerce. Calling recommendations the secret sauce for OTT businesses, Reed Hastings, co-founder of Netflix, quoted, "***If the Starbucks secret is a smile when you get your latte… ours is that the Web site adapts to the individual's taste.***" The content based model and collaborative model have been the two most evolved mechanisms that can be used to build the recommendation engines.[3] This paper uses the popular Movielens dataset. The dataset consists of movie data, user data, and the user ratings. It contains over 6040 users, 3883 movies and over a million user ratings.[4] The paper factors for the fundamentals of recommendations by using the genre as the deciding factor for recommendations. The paper takes a leap ahead by using the collaborative model to understand user's past ratings and determining users who match his style of movie watching and rating. The blend of two models creates a unique recommendation system to reduce the time for decision making.

II. Exploratory Data Analysis

Our literature review points to the fact that genre is the most critical component when it comes to movie recommendations. Amongst the extensive exploratory analysis that we performed, the ones that stood out and attested our readings were the genre data analysis. We conducted genre analysis for people around the age group of 30 and compared the same with younger audiences of age below 18. Comedy, Drama and Action came out as the top 3 genres with Comedy being the undisputed winner irrespective of age groups. The gender wise break up as shown in "Fig. 1" and "Fig. 2", manifest how genres matter irrespective of the genders. With age comes drama, is what the data suggests as drama and action interchange their ranks in the two segments analysed. On the other end of spectrum, are film-Noir and documentary, as there is a very small audience for the same. That brings out an interesting insight which suggests that there is potentially a user segment with similar interests in them. The word cloud "Fig. 3" affirms that Comedy and Drama are the most watched.

III. Content Based Model

The content based model is a learning algorithm that uses key features of a product or service to determine recommendation. It uses keyword analysis for classification. The algorithm uses TF-IDF (Term Frequency- Inverse Document Frequency) vectorization to find the most significant words in the document. TF is the frequency of a word in the document and IDF is the weight of the word based on how commonly it is used. Words occurring frequently are given lower weights as they are of lesser importance.

$$\textbf{TF-IDF score:}\quad w_{ij} = TF_{ij} \times IDF_i$$

$$TF(i,j) = \frac{\text{Term i frequency in document j}}{\text{Total words in document j}}$$

$$IDF(i) = \log_2\left(\frac{\text{Total documents}}{\text{documents with term i}}\right)$$

Fig. 5: Formula for TF-IDF score calculation

The TF-IDF value thus considers the frequency and weight of the words to give the most significant words. Our exploratory data analysis establishes the fact that genre of the movie is the keyword to be used for the TF-IDF calculation. An assumption we make here is that movies of a certain genre have similar content.

The algorithm starts with separating the genre string in the movie data into an array of strings. The TfidfVectorizer for English words is used for transformation over the array of genres. The resultant vector is saved in a variable named "tfidf_matrix". On studying the tfidf_matrix, we found there are 127 unique words in the genre column and the most significant words are drama thriller, thriller, drama, western, and children drama.

Next we calculate the cosine similarity. It is the angle between two vectors in a multi-dimensional space. Cosine similarity is proportional to the dot product of two vectors and inversely proportional to the product of their magnitudes and that is why we use cosine similarity instead of Euclidean distance. The angle between the vectors determines how similar they are. Smaller angle means higher similarity. Then the array of cosine similarities is to be created. To do this, we compared the performance of cosine_similarity and linear_kernel methods to calculate the cosine similarity and linear_kernel approach was average 30% faster. The array created from linear_kernel is used to calculate the top 10 movies. This gives us the top 10 rated movies from the genre of the input movie. The output is some of the cult movies of the corresponding genre. The model's limitation though is that it assumes genre is the deciding factor for recommendation leading to the same set of movies in the output for a given genre. Hence, we proceed to the next model which uses the collaborative approach to give a personalised touch to the recommendations.

IV. Collaborative filtering

Collaborative filtering leverages a user's consumption history to generate the recommendations. Similarities between users or the items, or both the users and items can be used in this model. This way the model adds a personalized touch to the output and there is a high likelihood of the user consuming the recommended content.

In this model, we derive the top 10 movies by finding top movies as per the ratings of users who have rated movies similar to the user for whom we are recommending the movies. First, we create a sample user with predefined ratings for 5 movies. We then derive the users who rated the above movies. For these users we derive the ratings for the movies that our sample user has rated. Next we identify users who are similar to our sample user by calculating the Pearson Correlation Coefficient or Centered Cosine Similarity. [4] Pearson Correlation coefficient gives the magnitude and direction of similarity between the points. Thus it tells us how closely related the users are.

We use the top 50 users based on the highest value of PC coefficient and analyse the movies rated by these users. The user ratings are converted to a weighted rating by multiplying it with the similarity index. This ensures that the users' ratings are only as important as their similarity to the sample user. Next step is to calculate identify the top movies from the above set of movies. To get the top movies we calculate the sum of weighted ratings for the given movie and divide it by the sum of similarity index. The top 10 movies with highest average recommendation scores are returned as the result for recommended movies.

V. VALIDATION USING RMSE

To validate our collaborative model, we calculate RMSE and compare the outputs. The RMSE is calculated for both, user-user and user-item based collaborative models. For calculating RMSE, we take 2% of data as sample and then divide it into train and test sets. Predicted ratings are calculated using the Pearson coefficient by using user-user and user-item filtering approach. RMSE is calculated for difference in actual and predicted ratings. It is observed that the RMSE for user-based model is 1416 and for item-based model is 1636. Thus, user-based model looks to be more effective.

VI. CONCLUSION AND FUTURE WORK

We explored the three available approaches of recommendation models. As per our analysis, the content-based model is recommended for new users. The collaborative approaches get a cut above the content based when there is significant user and ratings history available in the system. From the two collaborative methods, the user-user collaborative filtering is better suited when the users ages in the system and we have ratings to understand his likes and dislikes. As next steps, we would like to explore the possibility of extending the recommendations to be a combination of our models depending on location of a user and the time at which a user is watching a movie.

VII. ACKNOWLEDGEMENTS

We thank our Professor Dr. Nidhi Rastogi and TA Rigved Rakshit for the continued guidance during the project. We also want to acknowledge the silent but invaluable inputs we received from our references to strengthen our understanding of the unventured territories.

VIII. Work Plannning

The project was divided into 5 sub tasks and each member took up the ownership of each task. The workload was shared equally by everyone and was reviewed every week. The project was planned to be completed within 14 weeks. The daily communication and commitments for finishing tasks were documented on Discord. Meetings were planned to be conducted bi-weekly for checking the progress. The task break up was as shown in "Fig. 4"

IX. INDIVIDUAL CONTRIBUTION

My contribution towards this project was mainly in Exploratory Data Analysis (EDA) and working on content-based and user-based collaborative recommendation model.

EDA is an important step in determining what key features should be used in model building. I helped visualize what genre is the most popular among different genders and within younger audience. I had also worked on finding the frequency of genres occurring in this dataset and also the distribution of the rating present in the data set. With the help of word-cloud, we were able to identify the most popular genre and the other way around. We had also plotted to visualize the number of occurrences for the same. Rating analysis was also one of the most important components of the EDA as it was the key feature we used in the CF user-based model. Because of this, we were able to find that user ratings were biased and the most popular rating was around 4. This observation was also one of the reasons why we decided Pearson-correlation was the best approach to handle "tough raters" as "easy raters". Our visualizations using the feature occupation also helped us understand how the rating and count of ratings differed with different occupations.

From the work we did in EDA, we were able to identify genre as the key feature that should be used in the content-based model. With the numerous resources online, I was able to understand how tf-idf vectorizer is used and the importance of it. Using the tf-idf score, I helped identify which similarity metric could be used for content-based. In the end, we were able to determine how linear kernel method is 30% more time efficient on average than cosine similarity and we decided to go ahead with this approach for the recommendation.

For the user-based collaborative filtering model, after referring various research papers and resources online, we decided that the user-rating will be the best feature in our dataset with which we can implement it. One of the key challenges that we faced while working was identifying the best fit similarity metric. During the literature survey, I came across different similarity metrics that were generally used in a CF recommender system. Our first approach was using Jaccard Similarity. The problem with this was that it takes into account only how many movies two users have rated in common and not by how much the user prefers the movie. Our second approach (Cosine similarity) also had few drawbacks with respect to filling the missing rating values to compute the cosine value. The final approach we implemented was the Centered-cosine (Pearson coefficient) which helped overcome the disadvantages of the previous two approaches. By using this method, the average rating value was centered around 0 and using the normalized vector, we were able to compute the cosine similarity to find similar users.

Overall, through our discussions on Discord and the weekly meetings we had through Zoom helped us work together and resulted in a good learning experience for me.

REFERENCES

[1] statista.com, 'OTT video revenue worldwide from 2010 to 2026',2021. [Online]. Available: https://www.statista.com/statistics/260179/over-the-top-revenue-worldwide. [Accessed: 21- Nov- 2021].

[2] ericsson.com, 'TV and Media 2017', 2017. [Online]. Available: https://www.ericsson.com/en/reports-and-papers/consumerlab/reports/tv-and-media-2017. [Accessed: 21- Nov-2021].

[3] towardsdatascience.com, 'introduction to recommender systems', 2020. [Online] Available: https://towardsdatascience.com/introduction-to-recommender-systems-1-971bd274f421, [Accessed: 20- Nov- 2021]

[4] F. Maxwell Harper and Joseph A. Konstan. 2015. The MovieLens Datasets: History and Context. ACM Transactions on Interactive Intelligent Systems (TiiS) 5, 4, Article 19 (December 2015), 19 pages. DOI=http://dx.doi.org/10.1145/2827872

[5] moonbooks.org, 'How to calculate the Pearsons Correlation coefficient between two datasets in python, 2021. [Online] Available: https://moonbooks.org/Articles/How-to-calculate-the-Pearsons-Correlation-coefficient-between-two-datasets-in-python-/. [Accessed: 28- Nov- 2021]

FIGURES:

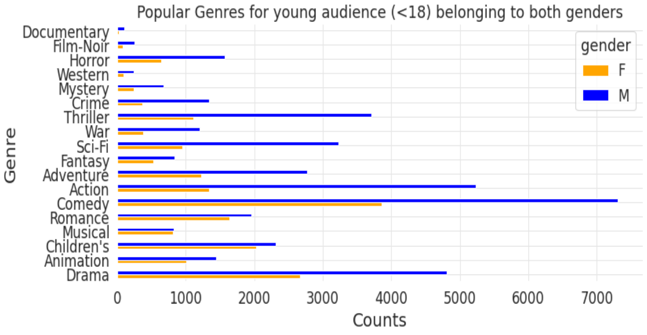


Fig. 1. Popular Genres for young audience (<18) belonging to both genders

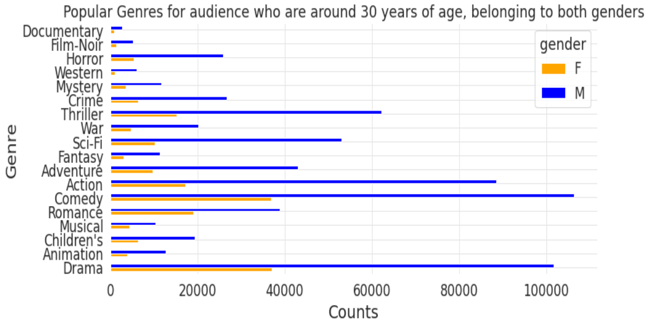


Fig. 2. Popular Genres for audience who are around 30 years of age, belonging to both genders

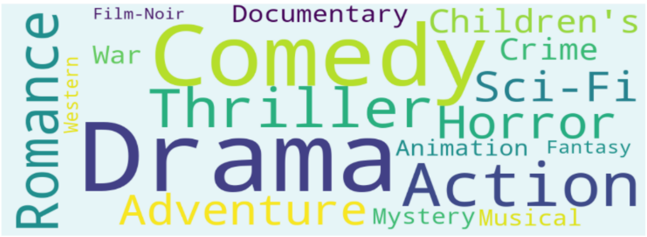
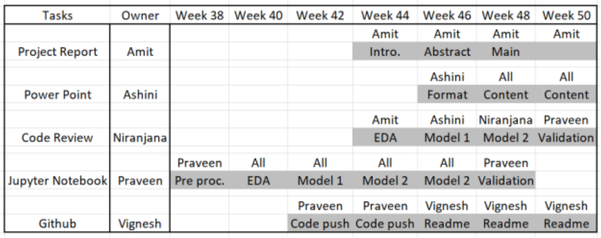


Fig. 3. Genre Word cloud



Tasks	Owner	Week 38	Week 40	Week 42	Week 44	Week 46	Week 48	Week 50
Project Report	Amit				Amit Intro.	Amit Abstract	Amit Main	Amit
Power Point	Ashini					Ashini Format	All Content	All Content
Code Review	Niranjana				Amit EDA	Ashini Model 1	Niranjana Model 2	Praveen Validation
Jupyter Notebook	Praveen	Praveen Pre proc.	All EDA	All Model 1	All Model 2	All Model 2	Praveen Validation	
Github	Vignesh			Praveen Code push	Praveen Code push	Vignesh Readme	Vignesh Readme	Vignesh Readme

Fig. 4. Gantt chart for project work