MOVIE RECOMMENDATION SYSTEM

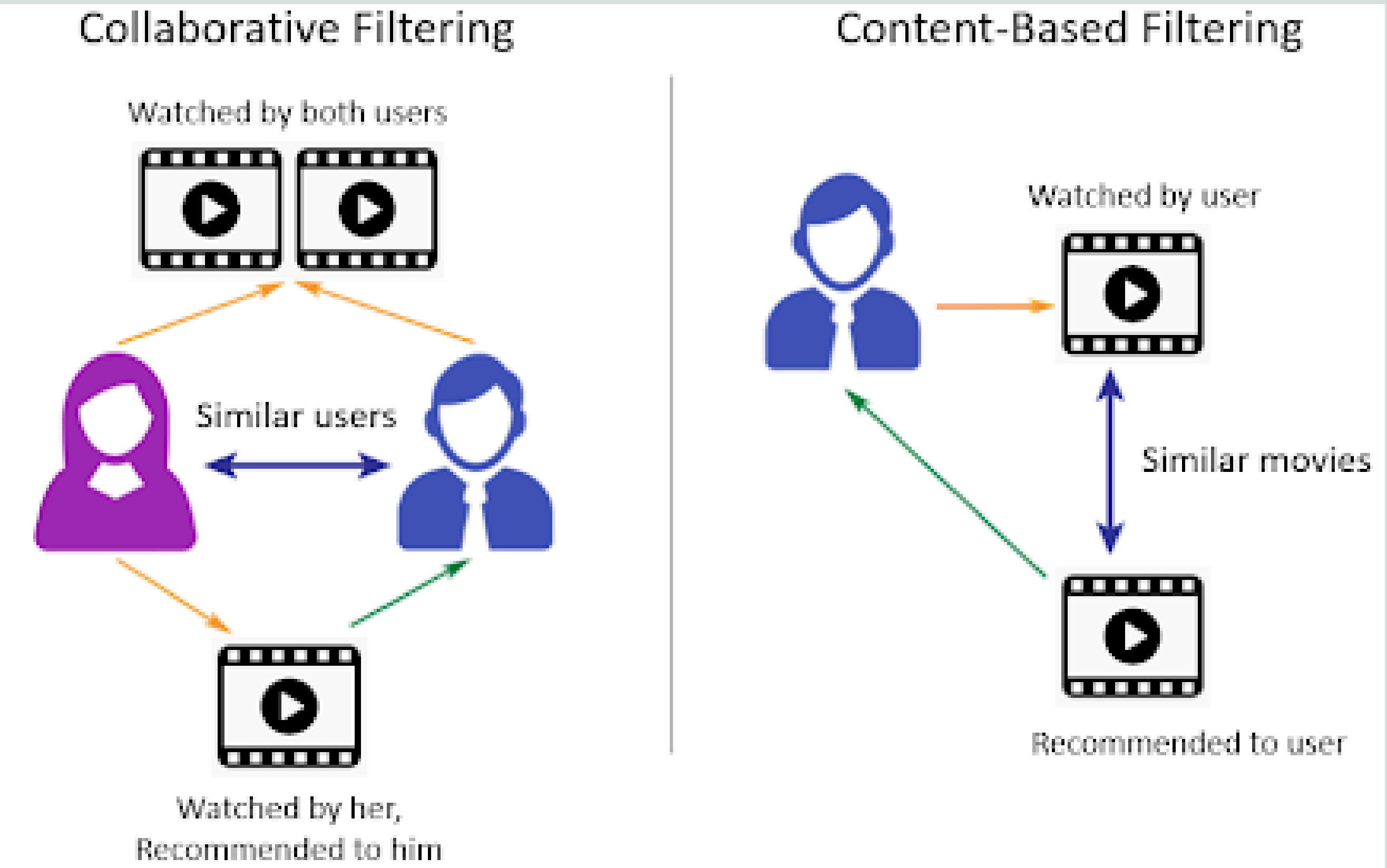
Ashna Sood, Urmi Suresh, Tae Kim, Xianglong Wang
University of California, San Diego
Data Science Student Society

BACKGROUND

- A recommendation system is an algorithm filtered to a target user that provides predicted ratings/suggestions to that user based on their previous preferences
- Goal: Create a movie recommender system that effectively evaluates the target user's previous predictions and ratings and outputs movies similar to the input movie.
- Started out with Netflix data which did not have enough metadata information to create an effective recommendation system
- Found a new MovieLens dataset that contained more comprehensive movie metadata

THE DATA & DATA CLEANING

- MovieLens Dataset
- 45,000 movies
- Movies released on or before July 2017
- Cast, crew, plot keywords, budget, revenue, posters, release dates, languages, production companies, countries, etc.
- 26 million ratings on a 1-5 scale from 270,000 users for all the movies
- extracted key features like director, top 5 crew, writer, executive producer, and most relevant key words, in addition to genre, collection



FUTURE WORK

- Create a movie recommender website where users can create a movie history profile by rating movies they liked/disliked and the the hybrid model will output suggestions factoring in both the user profile and movies with similar metadata
- Create a more rigorous machine learning neural network instead of calculating similarity matrices between movies and users
- Explore and further develop other forms of recommender systems such as restricted boltzmann machines
- Create a better metric evaluation system of the models

CONTENT BASED APPROACH

Content based recommendation systems use specific features of items to produce recommendations of other items that are similar to what a user has already indicated that they like. The user provided data is explicitly expressed via a rating the user has previously given, or implicitly expressed via clicking on a link. We created a cosine similarity matrix using a TF-IDF table of every pair of words in each one of the movies' metadata strings. We decided on the cosine similarity matrix after exploring other options because the cosine similarity is efficient at comparing how similar the contents of two vectors is and is minimally affected by what the magnitudes of the vectors are.

COLLABORATIVE FILTERING APPROACH

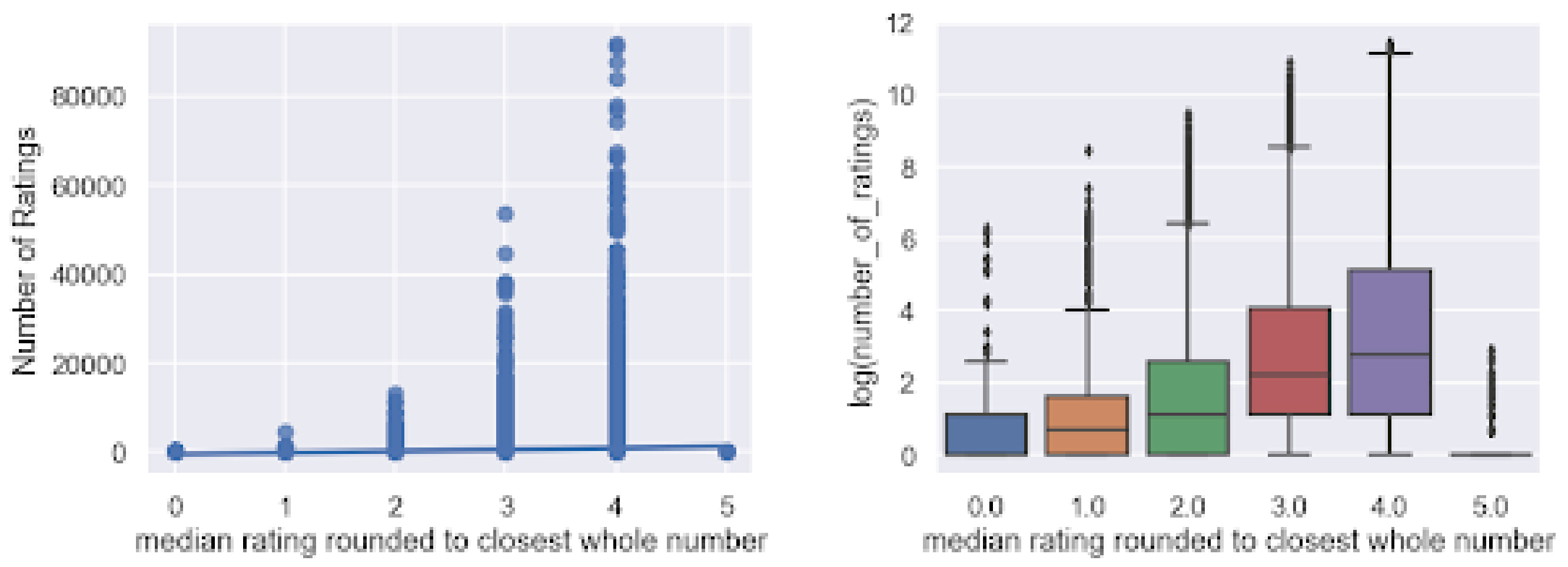
Collaborative filtering is an algorithm that creates recommendations based on data collected from other users, using the assumption that users who have similar interests in certain items are more likely to see eye to eye again. Our collaborative filtering algorithm uses the similarity index technique where a certain number of users are chosen based on how similar they are to the user we are focusing on. Then a weighted average of the selected users is created and that number provides the suggestions for the target user.

HYBRID RECOMMENDER

The Hybrid Recommender System combines the strengths of both the content-based and collaborative filtering models. Using both the metadata features of the input movies and the user's past preferences and predicted ratings of those movies, the model curates more accurate recommendations. The content-based model calculates the cosine similarity between the movies and the 20 most similar movies are fed into the collaborative filtering model that predicts and ranks the user's ratings for each movie in descending order. The returned recommendations are ideal as it incorporates both user preferences and movie feature similarities.

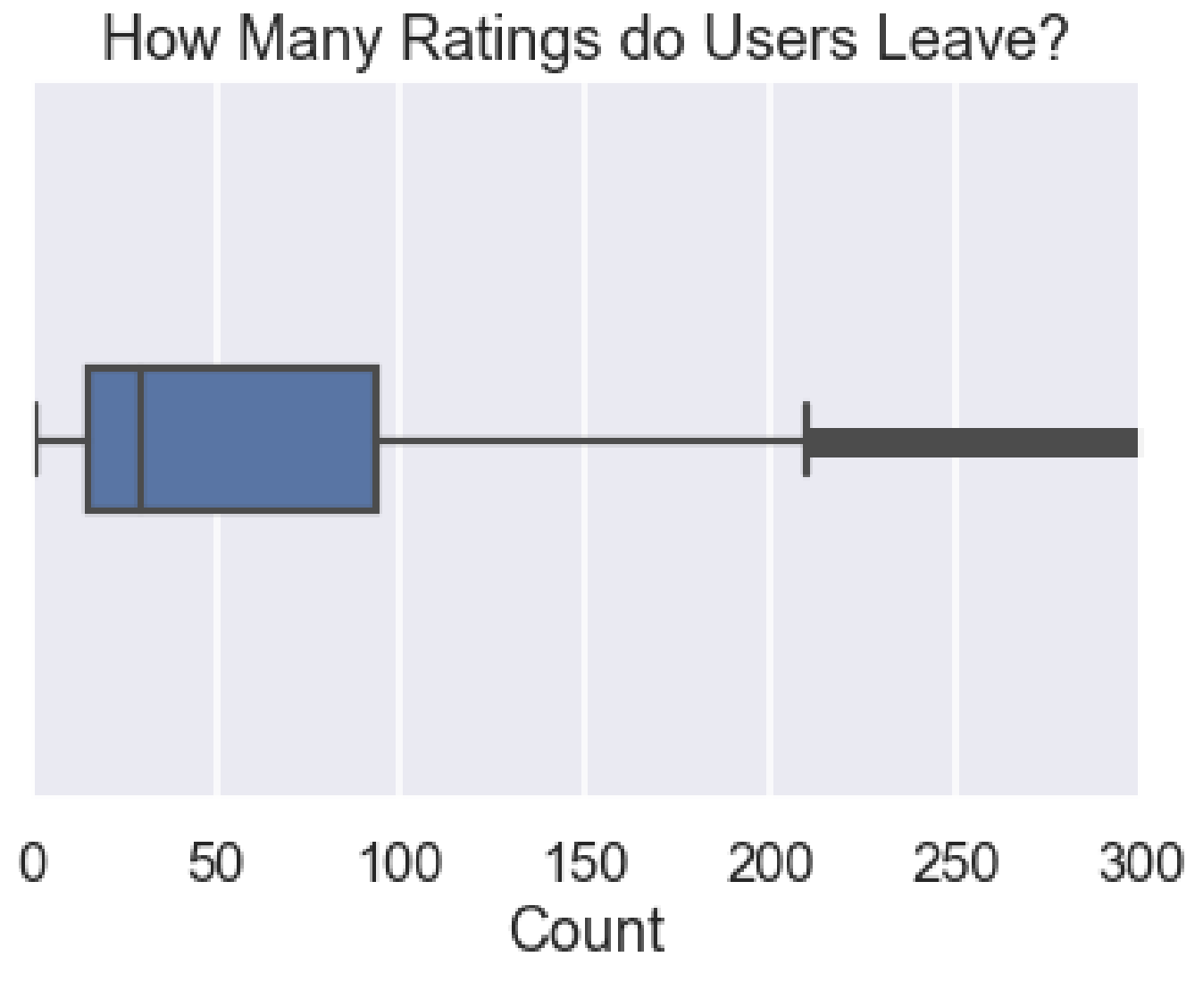
EDA

Do Movies With Higher Median Ratings Have More Number of Ratings?



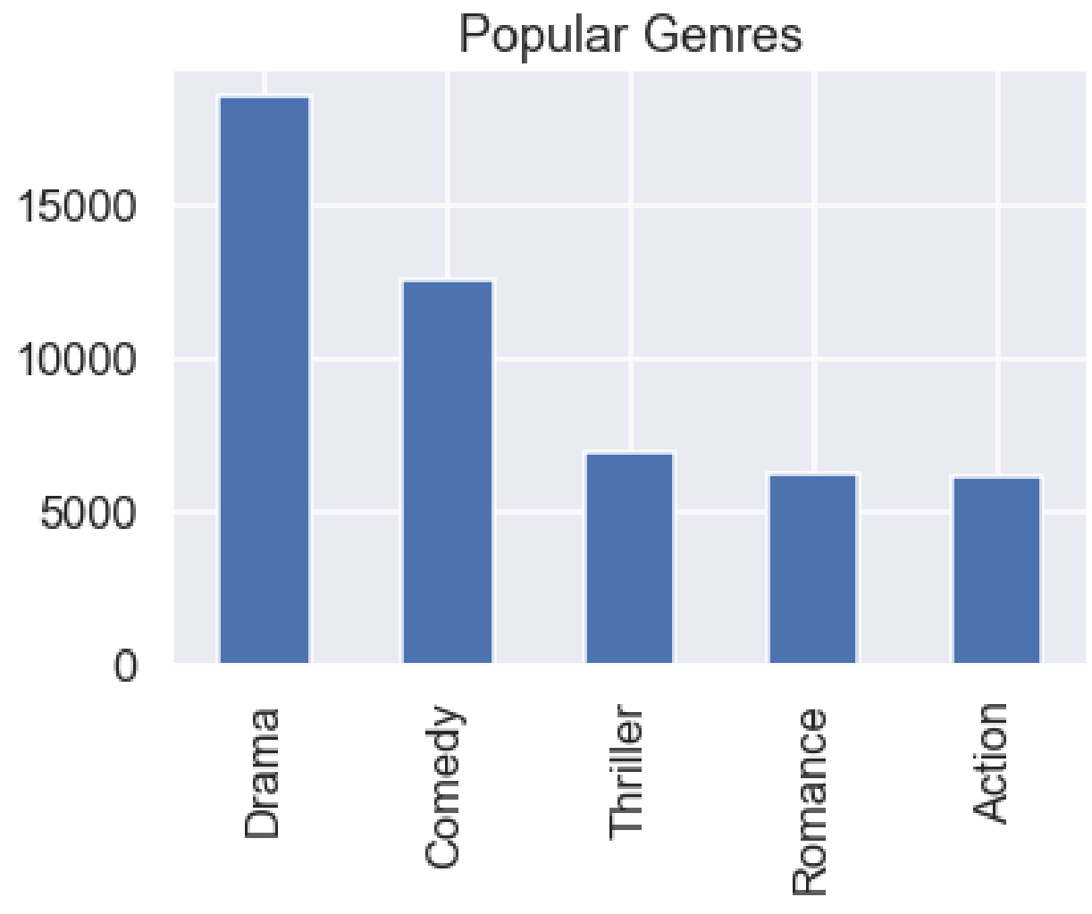
From the scatterplot and boxplot above we see that movies with higher median ratings tend to have higher number of people who have rated and viewed those movies as they are probably more popular movies

How Many Ratings do Users Leave?



The boxplot distribution shows on average how many ratings a user leaves. The average number seems to be around 30 ratings.

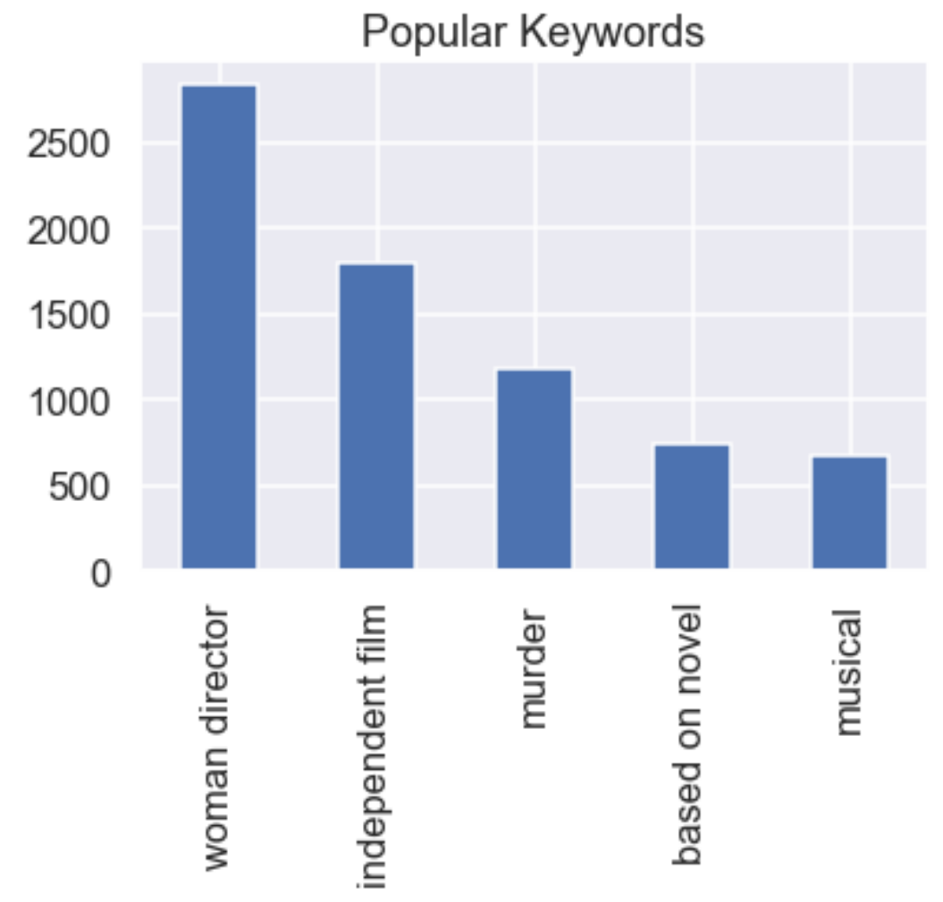
Popular Genres



In the graph to the left we can that Drama was the most frequently appearing genre in the dataset followed by Comedy, Thriller, Romance, and Action.

In the graph to the right we can see that the most popular keyword in the dataset was "woman director" followed by "independent film" and "murder".

Popular Keywords



DEMOS & TESTING & RESULTS

Created 3 user profiles (Romantic Comedies, Action, Bollywood) with watch history and ratings

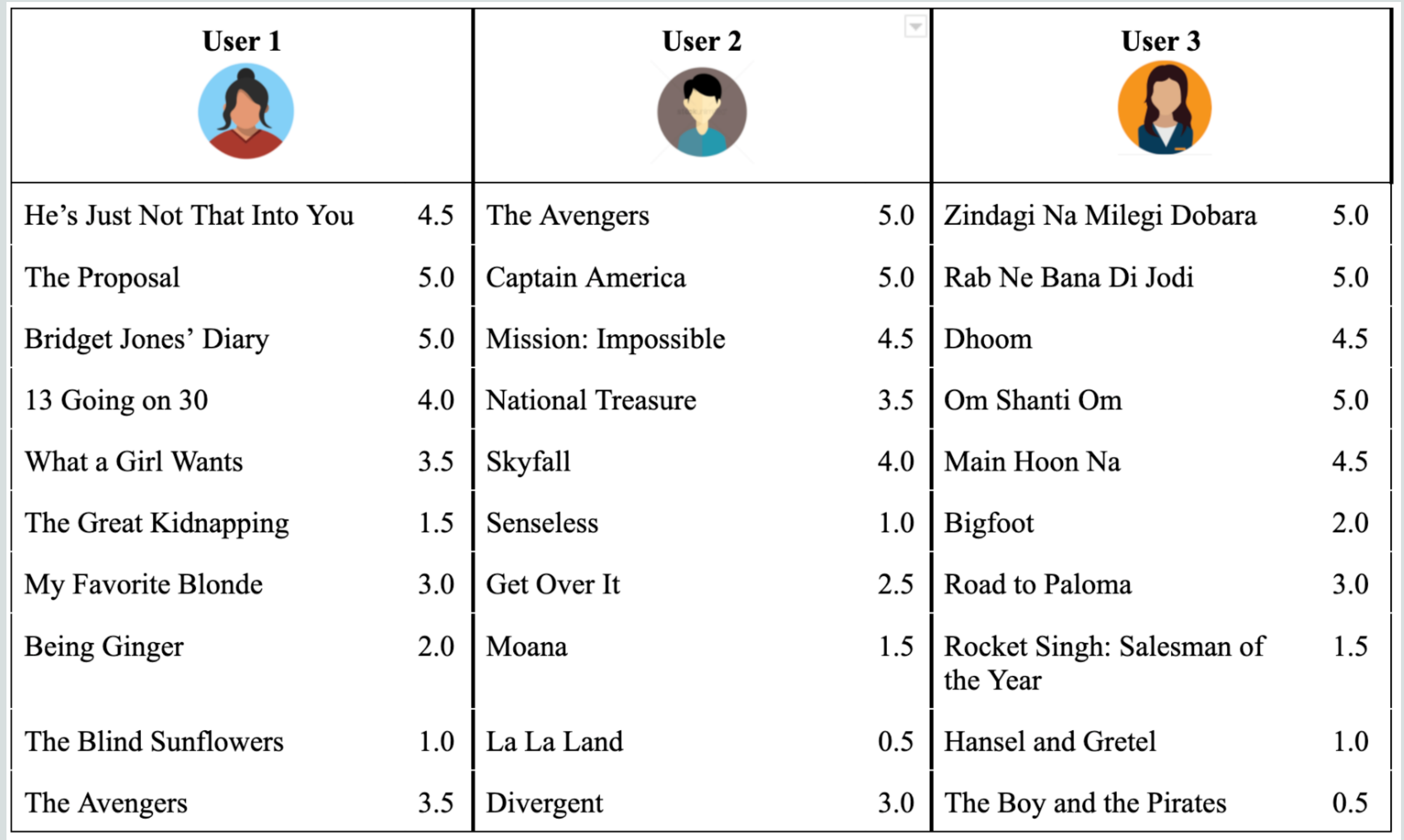
- Re-calculated similarity scores between new users and existing dataset
- Hybrid model suggested movies curated to each user profile and based on movies metadata

Performed 2 tests to evaluate and compare each recommender model we created:

1. Generated 3 lists of 20 randomly selected movies from dataset to feed into each model to evaluate the performance of the different approaches
2. Fed same 7 movies from various genres to each model

Results:

- Content-based is good at generating movies in the same collection and genre with similar actors/directors
- Collaborative filtering is good at providing accurate ratings to movies that other users with similar movie preferences rated
- Hybrid model considers context of movies while also accounting for user preferences and predicted ratings for that movie, combining the strengths of both models



User 1		User 2		User 3	
He's Just Not That Into You	4.5	The Avengers	5.0	Zindagi Na Milegi Dobara	5.0
The Proposal	5.0	Captain America	5.0	Rab Ne Bana Di Jodi	5.0
Bridget Jones' Diary	5.0	Mission: Impossible	4.5	Dhoom	4.5
13 Going on 30	4.0	National Treasure	3.5	Om Shanti Om	5.0
What a Girl Wants	3.5	Skyfall	4.0	Main Hoon Na	4.5
The Great Kidnapping	1.5	Senseless	1.0	Bigfoot	2.0
My Favorite Blonde	3.0	Get Over It	2.5	Road to Paloma	3.0
Being Ginger	2.0	Moana	1.5	Rocket Singh: Salesman of the Year	1.5
The Blind Sunflowers	1.0	La La Land	0.5	Hansel and Gretel	1.0
The Avengers	3.5	Divergent	3.0	The Boy and the Pirates	0.5

METRICS

Evaluating recommender systems is challenging as it is hard to curate actual predictions that users will find to be suitable. Different types of recommender systems require different metrics, and some don't have a scoring metric. For example, content-based recommendation systems cannot be scored because there is no such data set that objectively shows how similar two movies are. However, the collaborative filtering model is evaluated using RMSE and MAE and can compare the error between actual user ratings and predicted ratings. The hybrid model is scored using RMSE. The method of scoring is as follows:

- Black out some of the ratings in the test set
- Have the hybrid model predict the ratings for each of the blacked-out movies.
- Calculate the RMSE and MAE of all movie ratings that were initially blacked out by comparing the actual rating to the predicted rating.

RMSE	0.7961
MAE	0.6022

The RMSE value indicates that the model predicts the user ratings relatively well. The MAE value tells us that the difference between the actual and predicted values is minimal.

GITHUB LINK:
HTTPS://GITHUB.COM/ASHNASOOD/DS3_MOVIERECCOMENDERSYSTEM