Age and Context Sensitive Entertainment Recommendation Model

Machine Learning Final semester project evaluation

INDRAPRASTHA INSTITUTE *of* INFORMATION TECHNOLOGY **DELHI**

Motivation

As fun and easy as it is to binge your favorite shows with your loved ones, choosing what to watch is a challenging task. With so many genres to choose from, different preferences among individuals, and choosing age-appropriate content, narrowing down on the top movies and shows can be complicated. With our project, we aim to create a movie recommendation system that considers the family members' preferences, past activities, and age and recommends an age-appropriate movie for the family to watch.

Literature Review

1. User Based vs Item Based Filtering [1]

In user based filtering, interests of users were compared to find similar users.
Pearson Correlation Coefficient was used to determine the similarity between ratings. The higher the coefficient, the more correlated were the two users. An algorithm which found N nearest neighbors and made clusters of neighbors was used for user based Filtering. Since ratings change from time-to-time, this method is not static. In Item Based Filtering, Log Likelihood similarity was used. User ratings once given do not change, hence values of similarity would remain constant.

2. Movie recommendation system using collaborative learning [2]

In this paper, a collaborative approach is used in prescribing movies to others with similar tastes, allowing users to explore more. A web application is implemented that will enable users to rate movies and recommend appropriate movies based on others' ratings. This paper concluded that the content-based recommendation systems work on individual users’ ratings, limiting users from exploring more. However, the collaborative learning approach computes the connection between different users and, relying upon their ratings, recommends movies to others with similar tastes, allowing users to explore more.

Dataset Description

Dataset Description For Content and Demographic Filtering:

Our dataset contains the following attributes:

- type: If it is a movie or a TV show.
- title: The current title of the movie.
- director: The director(s) of the movie/TV show.
- cast: The actor(s)/actress(es) of the movie/TV show.
- country: The country in which the movie was released.
- date added: The date on which the movie/TV show was released.
- age_rating: The age rating given to the movie/TV show.
- duration: Total duration of the movie/TV show
- listed_in: Genre(s) of the movie/TV show.
- description: Sentence(s) describing the movie/TV show.
- IMDB: The IMDB rating of the movies/TV show
- TMBD: The tmdb rating of the movies/TV show

Dataset Description For Collaborative Filtering:

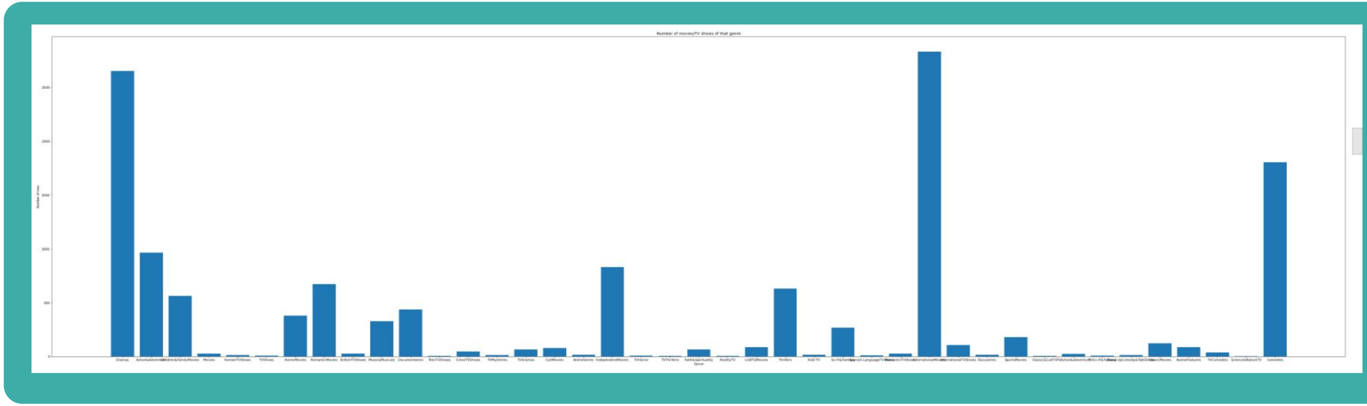
Our dataset contains the following attributes:

- title: The current title of the movie.
- user id: Uniquely identifies the user
- movie id: Uniquely identifies the movie.
- rating: The rating given by the user to a specific movie.

Data Preprocessing

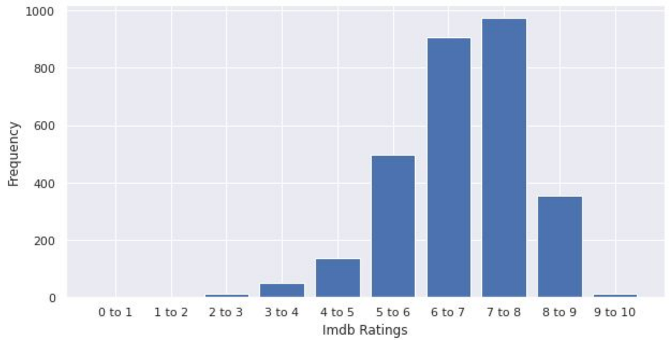
- Merging of Datasets
  - Two datasets were merged to generate a dataset of 10k+ rows.
  - Outer-join was performed.
- Removal of duplicate rows whilst maintaining max data
  - Duplicate rows due to slight difference /null values in columns of same title rows.
  - Information from all rows combined in a single row and dropping of duplicates.
- Feature Selection
  - Seasons, duration and director column dropped.
- Removal of empty values
  - Dropping of rows where value is empty.
- One - Hot Encoding of Genre
  - Listed_in consisted of multiple genres for movie/show. One-hot encoded the genres.

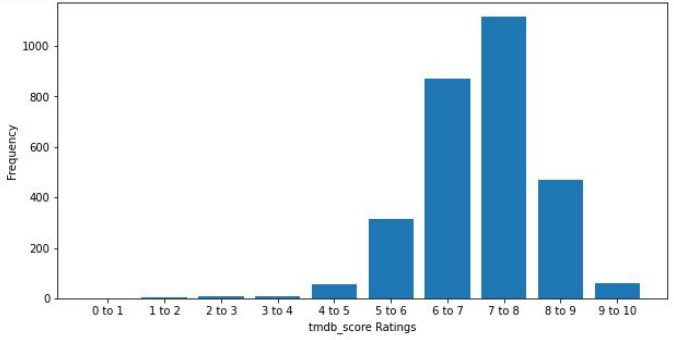
Exploratory Data Analysis or EDA



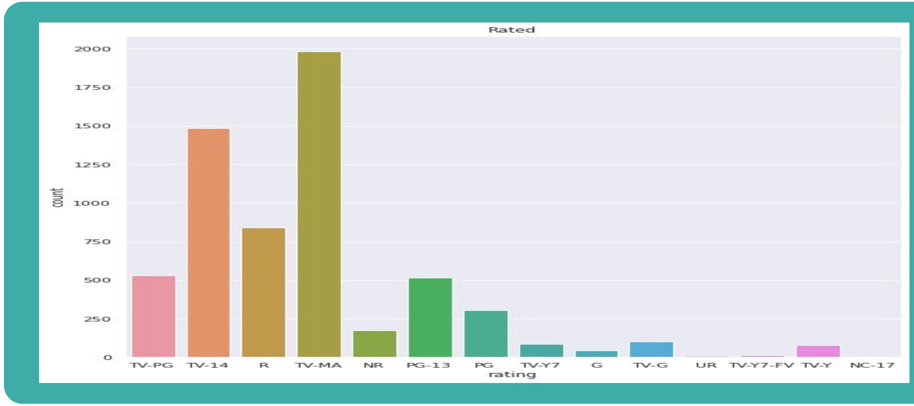
Genres of movies and TV shows. The highest genre comes out to be international movies, the second highest genre is Dramas

Frequency of the imdb and tmdb ratings of the movies and TV shows

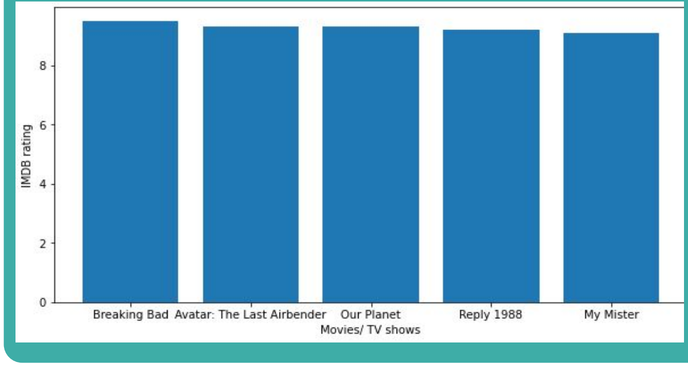




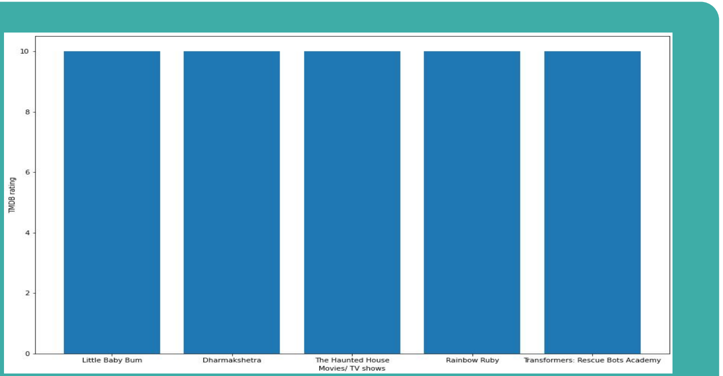
Exploratory Data Analysis or EDA



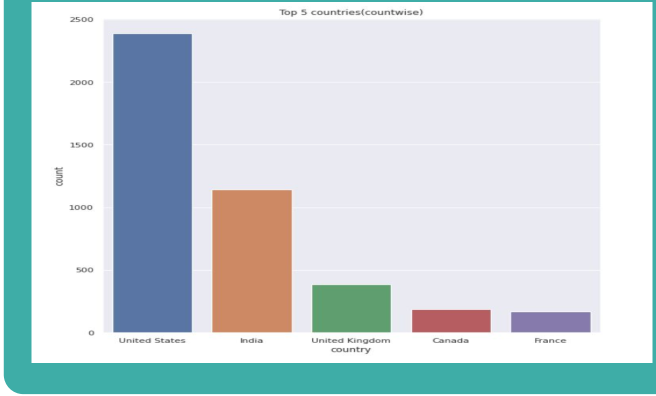
Count of the age certification ratings of the movies/TV shows of the dataset. TV shows with rating TV-MA(Mature Audience) have the highest count.



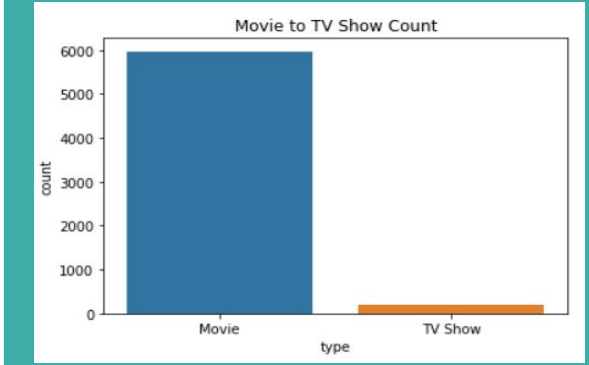
The Top 5 movies/Tv shows with the highest imdb rating(left) and tmdb rating(right)



Exploratory Data Analysis or EDA

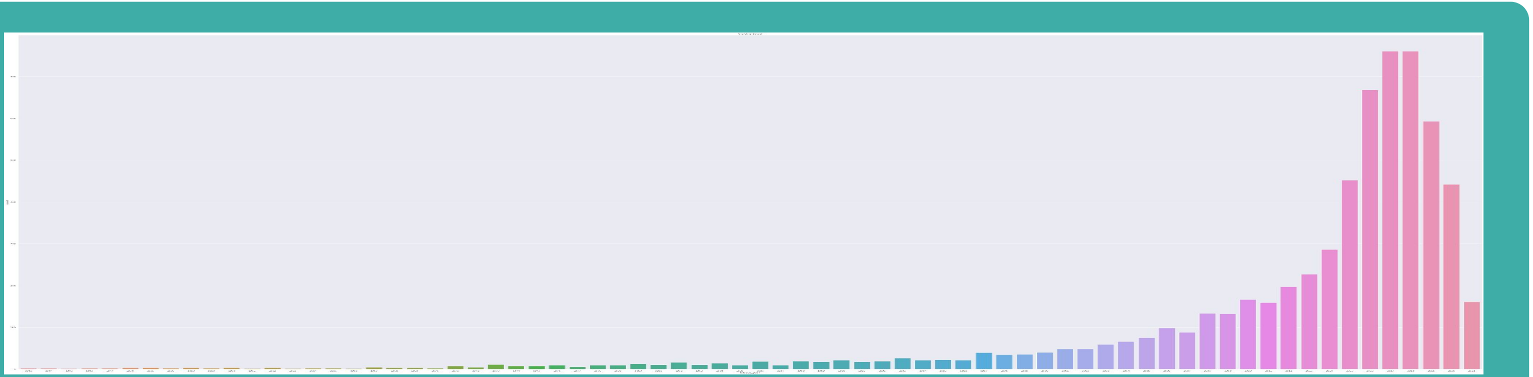


Countries that made the most movies/TV shows. USA is the highest among all the

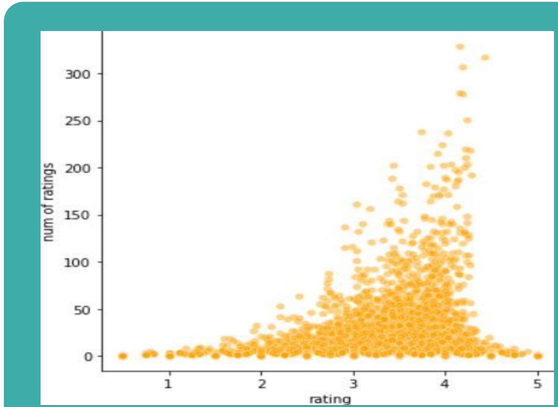


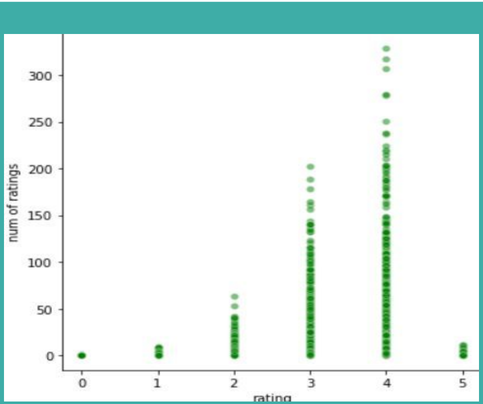
The count of movies and TV shows in the dataset. There are more movies than TV shows in our dataset.

Number of movies released in a year with max in 2016 and 2017



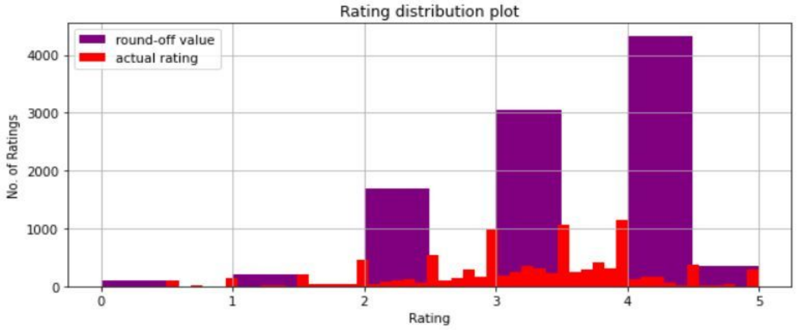
Exploratory Data Analysis or EDA





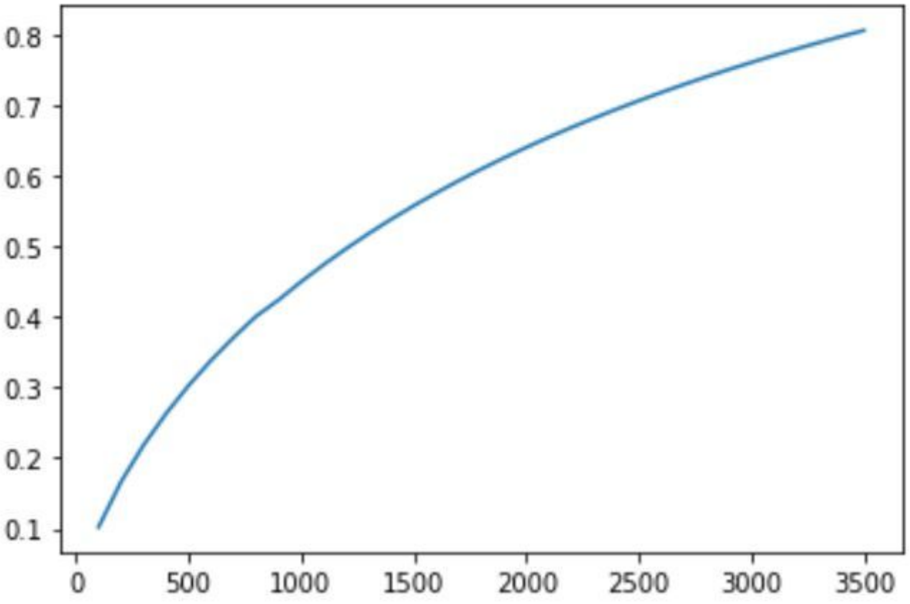
Scatter plot depicting for each movie its average rating(x) vs the number of ratings given by the user

A bell shaped plot depicting the ratings and round-off rating given by the users



PCA vs Variance

- Our final TFIDF matrix came to be around of size 8809 X 18000
- We can’t apply PCA on the 8k rows as these represent the movies
- We have applied PCA on the 18k vector.
- We plotted n(dimensions column vector is reduced to) Vs Variance
- Variance indicates the amount of information preserved after dimensionality reduction



N components vs Variance

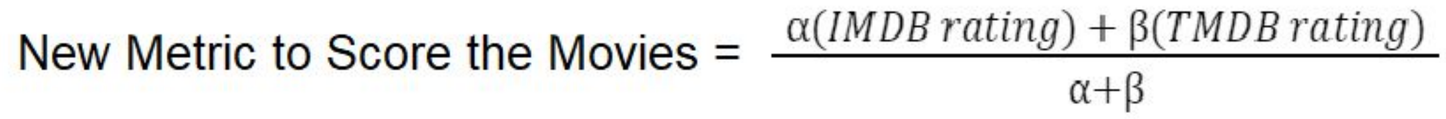
Methodology

Age Based Filters

We have different categories like TV-MA: strictly for adults, TV-14: for 14 and above, and TV-PG: for 14 and above but with parental guidance. Following this, we updated our dataset to remove ratings unsuitable for the given age group. We decide on suitable ratings by the min and max age present in the group. Min-age is calculated to remove adult movies, while max-age is calculated so PG movies(parental guidance) can be allowed when an adult accompanies the children.

Demographic Filtering

Demographic refers to the statistical characteristics of the human population. When a new user joins the system, the model is unaware of his past choices. This situation is termed a ‘Cold Start’. In such situations, we can predict the top-rated movies of all time to the user. In our dataset, we had access to both the IMDB and TMDB ratings of a movie. The Pearson correlation between them was found to be 0.57. This is a moderate-high positive correlation, i.e., if one increases, the other is also expected to increase.
The weights by which IMDB and TMDB ratings is multiplied are hyperparameters which can be tuned if real-time feedback is available.



$$\text{New Metric to Score the Movies} = \frac{\alpha(IMDB\ rating) + \beta(TMDB\ rating)}{\alpha+\beta}$$

IMDb ratings are more popular hence Imdb ratings were given a higher priority in the new_score calculation

Content Based Filtering

We calculated the Term Frequency-Inverse Document Frequency (TF-IDF) vectors of the description on an age filtered dataset. For quantifying similarity, we used Cosine Similarity as it is independent of magnitude and is much faster to calculate than the other two metrics without significantly affecting our final results.

The formal, mathematical definition of cosine similarity is as follows:

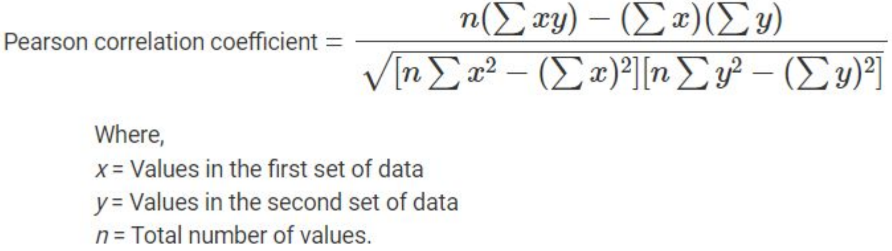
$$\text{cosine similarity} = S_C(A, B) := \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum\limits_{i=1}^{n} A_i B_i}{\sqrt{\sum\limits_{i=1}^{n} A_i^2} \sqrt{\sum\limits_{i=1}^{n} B_i^2}},$$

Finally, we sorted the list containing the pairwise cosine similarity of all the movies with the movie we chose and took the total sum of pairwise cosine similarity of each movie in the dataset with each movie of choice as our final similarity metric. We then recommended the movies in decreasing order of value of this metric.

Collaborative Filtering

In the collaborative filtering approach, we recommend the movies to a user based on other users having similar interests. We look for users with similar tastes and preferences to the current user. Then we recommend the movies favored by the matched users to the current user.

We make a pivot table with index value as the user id, columns value as the movie title and the feature whose statistical summary is to be seen is ratings. Now we input a movie name to which the current user wants similar recommendations. We calculate the Pearson correlation of the column vector of the input movie title with all other column vectors having ratings more than 20.



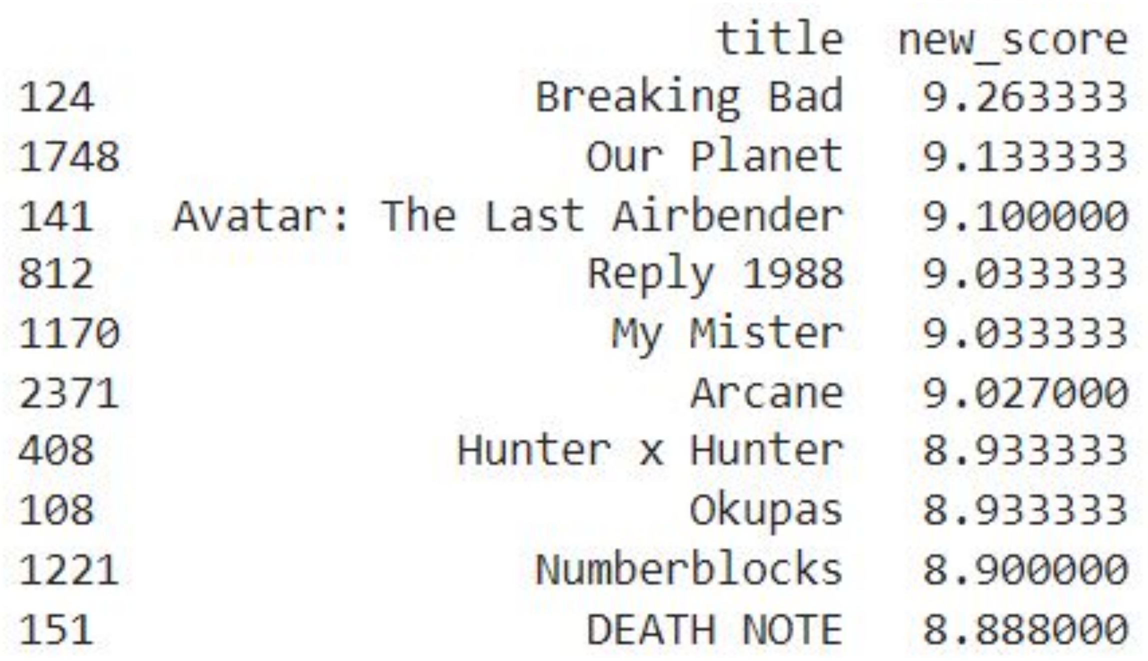
$$\text{Pearson correlation coefficient} = \frac{n(\sum xy) - (\sum x)(\sum y)}{\sqrt{[n\sum x^2 - (\sum x)^2][n\sum y^2 - (\sum y)^2]}}$$

Where,
x = Values in the first set of data
y = Values in the second set of data
n = Total number of values.

Now we get the pairwise Pearson Correlation value of every movie with the input movie. Now, we recommend five movies whose pairwise Pearson Correlation is the highest.

Results/Analysis/Conclusion

Demographic Filtering

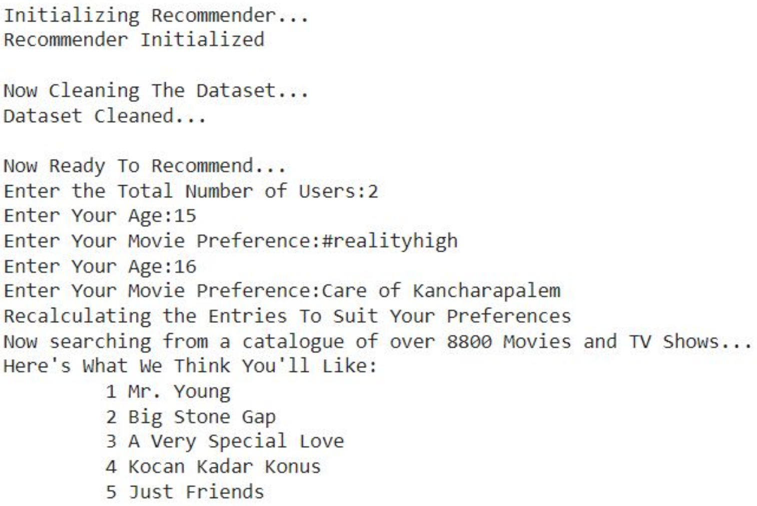


```
                               title  new_score
124                     Breaking Bad   9.263333
1748                      Our Planet   9.133333
141    Avatar: The Last Airbender   9.100000
812                      Reply 1988   9.033333
1170                      My Mister   9.033333
2371                         Arcane   9.027000
408                 Hunter x Hunter   8.933333
108                          Okupas   8.933333
1221                   Numberblocks   8.900000
151                      DEATH NOTE   8.888000
```

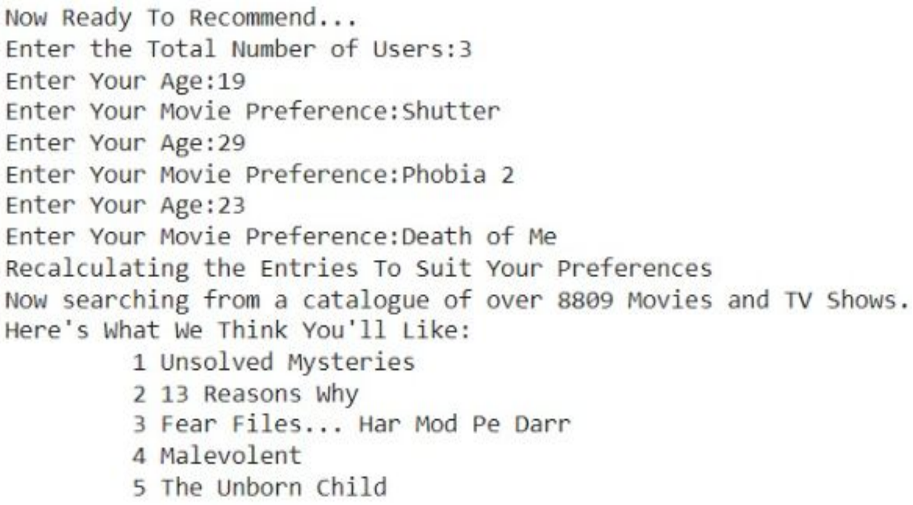
Movies with top scores are recommended

Results/Analysis/Conclusion

Content Based Filtering



```
Initializing Recommender...
Recommender Initialized

Now Cleaning The Dataset...
Dataset Cleaned...

Now Ready To Recommend...
Enter the Total Number of Users:2
Enter Your Age:15
Enter Your Movie Preference:#realityhigh
Enter Your Age:16
Enter Your Movie Preference:Care of Kancharapalem
Recalculating the Entries To Suit Your Preferences
Now searching from a catalogue of over 8800 Movies and TV Shows...
Here's What We Think You'll Like:
         1 Mr. Young
         2 Big Stone Gap
         3 A Very Special Love
         4 Kocan Kadar Konus
         5 Just Friends
```



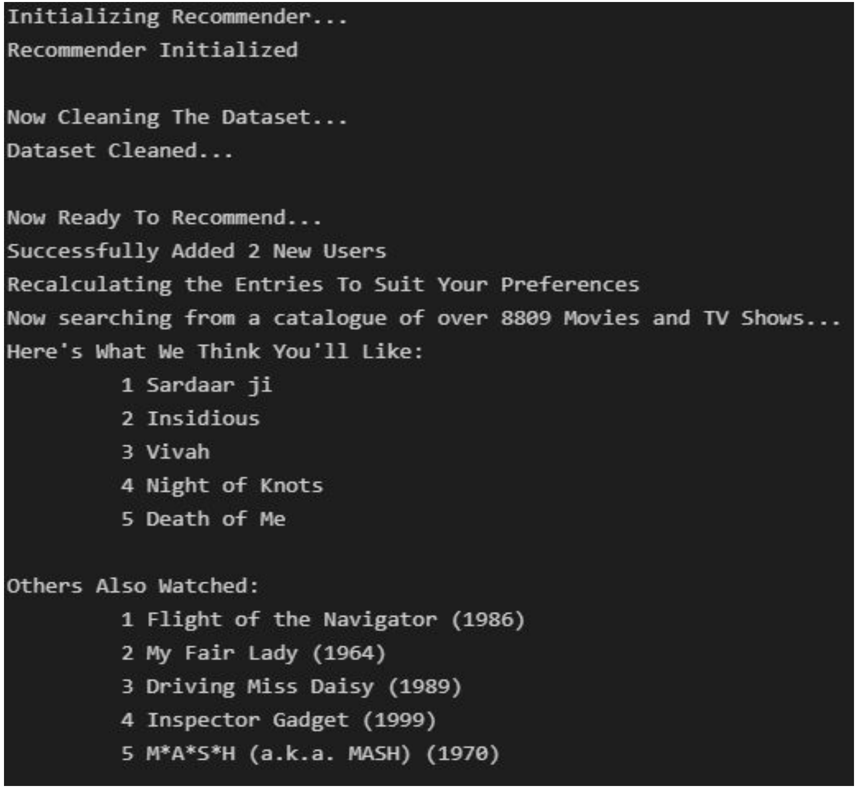
```
Now Ready To Recommend...
Enter the Total Number of Users:3
Enter Your Age:19
Enter Your Movie Preference:Shutter
Enter Your Age:29
Enter Your Movie Preference:Phobia 2
Enter Your Age:23
Enter Your Movie Preference:Death of Me
Recalculating the Entries To Suit Your Preferences
Now searching from a catalogue of over 8809 Movies and TV Shows.
Here's What We Think You'll Like:
         1 Unsolved Mysteries
         2 13 Reasons Why
         3 Fear Files... Har Mod Pe Darr
         4 Malevolent
         5 The Unborn Child
```

a) For a group of teenagers

b) For a group of adults

Results/Analysis/Conclusion

Collaborative Based Filtering



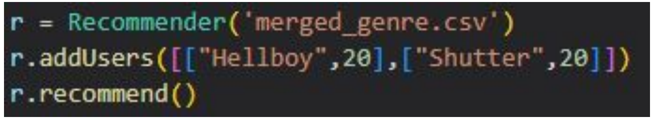
```
Initializing Recommender...
Recommender Initialized

Now Cleaning The Dataset...
Dataset Cleaned...

Now Ready To Recommend...
Successfully Added 2 New Users
Recalculating the Entries To Suit Your Preferences
Now searching from a catalogue of over 8809 Movies and TV Shows...
Here's What We Think You'll Like:
         1 Sardaar ji
         2 Insidious
         3 Vivah
         4 Night of Knots
         5 Death of Me

Others Also Watched:
         1 Flight of the Navigator (1986)
         2 My Fair Lady (1964)
         3 Driving Miss Daisy (1989)
         4 Inspector Gadget (1999)
         5 M*A*S*H (a.k.a. MASH) (1970)
```

Model Performance In Collaborative Filtering

- We cannot directly minimize the loss or calculate the performance metric of our model based on fixed data.
- True Performance can be analyzed by collecting user feedback for the recommendations made.
- It can be calculated through CTR. CTR is the number of times a recommendation is clicked by the user upon the total number of times the recommendation is shown.
- Over time, the CTR value of the model increases as it gains more information about the user preferences and choices.

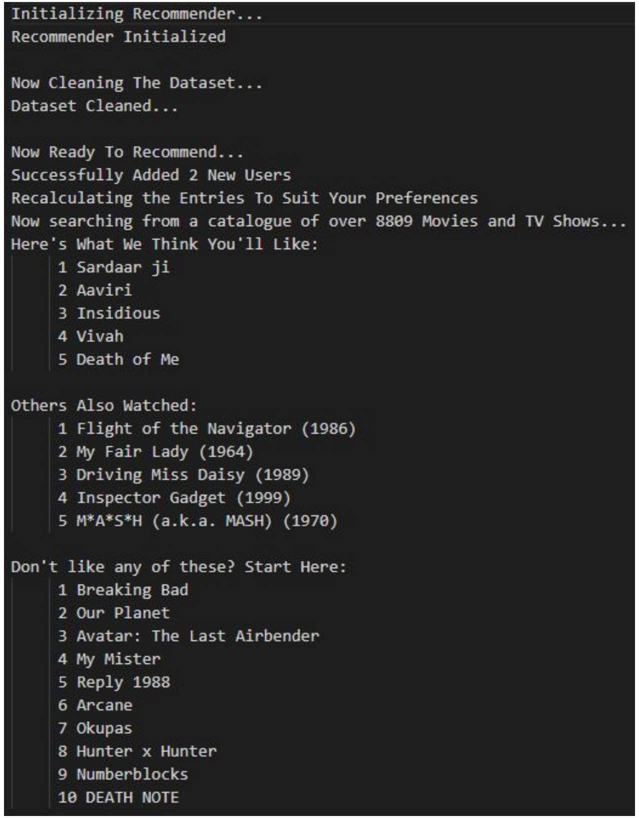
Combined Output

Input:



```
r = Recommender('merged_genre.csv')
r.addUsers([["Hellboy",20],["Shutter",20]])
r.recommend()
```

Output:

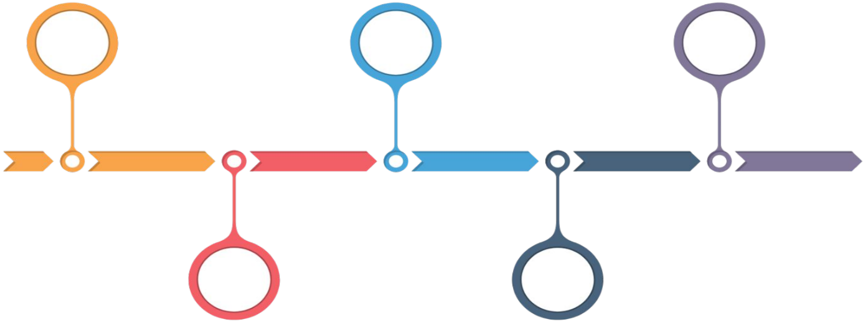


```
Initializing Recommender...
Recommender Initialized

Now Cleaning The Dataset...
Dataset Cleaned...

Now Ready To Recommend...
Successfully Added 2 New Users
Recalculating the Entries To Suit Your Preferences
Now searching from a catalogue of over 8809 Movies and TV Shows...
Here's What We Think You'll Like:
        1 Sardaar ji
        2 Aaviri
        3 Insidious
        4 Vivah
        5 Death of Me

Others Also Watched:
        1 Flight of the Navigator (1986)
        2 My Fair Lady (1964)
        3 Driving Miss Daisy (1989)
        4 Inspector Gadget (1999)
        5 M*A*S*H (a.k.a. MASH) (1970)

Don't like any of these? Start Here:
        1 Breaking Bad
        2 Our Planet
        3 Avatar: The Last Airbender
        4 My Mister
        5 Reply 1988
        6 Arcane
        7 Okupas
        8 Hunter x Hunter
        9 Numberblocks
        10 DEATH NOTE
```

Final Outputs

According to the timeline we proposed in the proposal, we have completed all the tasks that we proposed.
The tasks to be accomplished are as follows :

- Data Collection and Data Visualization
- Feature Analysis and Selection
- Principal Component Analysis
- Content Based Filtering
- Demographic Filtering
- Age Based Filtering
- Collaborative Filtering
- Model performance Analysis
- K - Means Clustering
- Hyperparameter Tuning
- Adding the option of entering more movie preferences per user in order to make a more accurate movie recommendation system.



Thank You!

Abhinn Yadav(2020013)
Dhruv Mishra(2020296)
Hunar Kaur(2020303)
Sadhvi Bhan(2020325)