# Netflix Analysis and Recommendation System

Dontneni Ganesh Sai
*University of North Texas*
Denton, USA
ganeshsaidontineni@my.unt.edu

Manohar Varma Buddharaju
*University of North Texas*
Denton, USA
manoharvarmabuddharaju@my.unt.edu

Sruthi Mullaguri
*University of North Texas*
Denton, USA
sruthimullaguri@my.unt.edu

Venkata Kavya Eti
*University of North Texas*
Denton, USA
venkatakavyaeti@my.unt.edu

***Abstract*—”Netflix Data Analysis,” a collaborative project, seeks to thoroughly examine a Netflix dataset that was obtained through Kaggle [1]. The dataset includes important parameters including show_id, type, title, director, cast, nation, date_added, release_year, rating, duration, listed_in, and description. After careful preparation, the group applies advanced exploratory data analysis (EDA) methods, looking at the distributions of individual variables, investigating the connections between features, and performing correlation studies to find complex patterns. Predictive modeling with possible feature selection is included in the project, including recommendation systems and user engagement prediction models. Model effectiveness is evaluated using evaluation measures such as F1 score, Precision, Recall, and Accuracy. The project's significance is seen in its ability to improve user experience, optimize content libraries, and refine content recommendations. The project intends to improve customer satisfaction and engagement in the ever-changing world of digital entertainment by utilizing EDA and predictive modeling to provide insightful information to the content sector.**

***Index Terms*—Exploratory Data Analysis (EDA), Predictive Modeling, Recommendation Systems**

## I. Introduction

### A. Motivation

In an industry dominated by online streaming services, Netflix's vast storage of user data is a treasure chest waiting to be discovered. This data, which goes beyond mere entertainment, holds significance for comprehending consumer preferences, viewing habits, and emerging trends. Uncovering trends within this massive dataset has the potential to transform content creation, improve user experience, and reshape the future of streaming. Exploring this enormous amount of Netflix data is one of the main goal of our project.

### B. Objectives

Our goal in deciding on an in- depth Exploratory Data Analysis (EDA) is to decode the complicated tapestry of patterns, trends, and user preferences established within the massive Netflix dataset. We seek to identify top directors, popular genres, and emerging trends that can shape content creation strategies through cautious examination. Beyond descriptive analysis, we hope to build predictive models capable of suggesting tailored content or anticipating user engagement patterns. We hope to offer practical suggestions to Netflix and other providers of content by leveraging the power of data-driven insights. These suggestions, which are based on an in-depth awareness of viewer behavior, may fuel strategic choices, optimize content libraries, and improve users' overall streaming experience.

## II. Related Work (Background)

The background and related work section would examine the broader landscape of data-driven approaches in the entertainment business with a particular focus on content streaming platforms, all within the context of Netflix data analysis. The literature that is now available emphasizes how important data analysis is becoming for improving platform optimization, content recommendations, and user experiences. This project is based on studies on user behavior, popularity of content, and personalized recommendations.

Additionally, to comprehend how algorithms support user involvement and content curation, research in the domains of collaborative filtering, predictive modeling, and recommendation systems is essential. Distinguished approaches including content-based filtering and collaborative filtering have been thoroughly investigated in the literature, offering valuable perspectives on the obstacles and prospects related to customized content recommendations.

It is possible to extract meaningful recommendations for Netflix and other content providers by combining EDA and predictive modeling, since this proposed technique offers a thorough insight of user preferences and content engagement patterns. The results of this study can help content producers improve user suggestions, optimize content libraries, and raise user engagement and satisfaction levels.

## III. Dataset

The Netflix-focused data set [1] that is being analyzed includes important parameters such as show_id, type (movie or TV show), title, director, cast, nation, date_added, release_year, rating, duration, listed_in (genres), and description. This data set provides the groundwork for in-depth investigation and the development of insights. Notably, pre-processing procedures were carried out, such as handling category variables and converting ”duration” to numeric values. A wide variety of information is captured in the collection, ranging from show-specific characteristics to more general information like release years and genres. Because of this richness, extensive exploratory data analysis (EDA) is performed to find patterns and trends in the Netflix content environment. The dataset also provides a comprehensive picture of user preferences and content dynamics on the Netflix platform by serving as the foundation for predictive modeling initiatives like developing recommendation systems and anticipating user involvement.

## IV. Design of Features

In designing the features for our Netflix data analysis project, we've taken a careful approach to gather information that helps us understand what people like to watch and how they engage with content. Imagine Netflix as a huge library with tons of shows and movies. Our features are like tools that help us dig into this library and discover interesting things.

Firstly, we look at basic details like whether a title is a TV show or a movie, the genres it falls into, when it was released, and how viewers rate it. These details give us a broad view of what's popular and when it became a hit. For instance, knowing the release year helps us spot trends over time.

Then, we dive into features that tell us how people engage with shows. How long is a show or movie? Do viewers prefer longer shows or shorter ones? How many seasons does a TV show have? We also look at when shows are added to Netflix – is there a pattern? Maybe certain months see more additions, and that could tell us about seasonal preferences.

To make our data more machine-friendly, we turn some words into numbers. For example, we convert whether a show is a TV show or movie into a number, and we do the same for ratings. This helps our computers understand and analyze the data better.

We love visuals too! We use pie charts, histograms, and bar plots to draw pictures of our data. These help us see the distribution of different types of shows, the release years, and popular genres. It's like creating colorful maps that guide us through the Netflix landscape.

Text is powerful too. We analyze show descriptions to find out what words pop up often. It's like studying the words on book covers to understand what the book might be about. These words give us clues about what themes and topics are popular among viewers.

For our tech-savvy side, we use a technique called collaborative filtering. It's a bit like looking at what other people who have similar tastes enjoy watching. We also apply some math magic called Singular Value Decomposition (SVD) to simplify complex patterns and make our analysis more efficient.

Finally, we want to predict how much viewers will like a show. To do this, we create a label that tells us if a show is engaging or not based on its rating. Then, we use smart computer models like Random Forest, Logistic Regression, and SVM to make predictions. It's like having a virtual assistant that learns from what people like and suggests similar shows.

Our design is like building a treasure map for the Netflix library, helping us discover hidden gems and understand what makes viewers click that 'play' button. It's not just about numbers; it's about turning data into stories about what we love to watch.

## V. Analysis

In the vast world of Netflix data, our goal is to uncover interesting stories and understand how people use the platform. Let's break down what we found:

**Content Overview:** Most of what you see on Netflix are movies, showing that people enjoy standalone films. The popular genres include Drama, Comedy, and Documentary, giving us a glimpse into viewers' diverse interests. Looking at release years, we notice a steady rise in content creation, especially during the 2010s.

**Viewer Patterns:** Understanding how users engage is crucial. We find that viewers prefer shows around 100 minutes long, and TV series with multiple seasons are quite popular. Adding content consistently throughout the year, regardless of seasons, indicates a steady viewer interest.

**Numbers Speak:** Turning words into numbers helps us understand better. Labeling ”Type” and ”Rating” helps us quantify them. A high number of movies and a wide range of ratings suggest a diverse content library.

**Visual Insights:** Pictures tell stories better. Pie charts show that movies make up about two-thirds of the content. Histograms track how content has evolved over the years. Bar plots display the dominance of Drama, Comedy, and Documentary genres, providing creators insights into what's popular.

**Words Matter:** Analyzing the words used in show descriptions is powerful. The word cloud highlights recurring themes like love and relationships, giving a sense of what resonates with viewers.

**Recommendation Magic:** Using collaborative filtering, we find shows similar to what viewers like, like getting recommendations from friends. SVD helps understand complex patterns, making suggestions more tailored to individual tastes.

In a nutshell, our analysis isn't just about numbers. It's a journey into what viewers prefer and how content evolves. It's about decoding the stories in the data, guiding content creators, and making the viewer's digital entertainment experience even better.

## VI. Implementation

Moving into the action phase, our goal is to turn the insights we found into practical steps, using a mix of data processing, visualizations, and machine learning methods.

**Data Cleanup:** First, we tidy up the data by dealing with missing info and converting things like 'Type' and 'Rating' into numbers. This helps us have a neat dataset ready for analysis.

**Visualizing Insights:** We bring our findings to life with visuals. Using tools like Matplotlib and Seaborn, we create charts to show how content is spread, trends over the years, and what genres people like.

**Predictive Modeling:**

Our approach combines collaborative filtering and dimensionality reduction techniques to optimize Netflix's content recommendation system. Recognizing the challenge of personalized suggestions in a vast content library, we utilize two distinct models—TF-IDF-based [6] linear kernel similarity and Truncated Singular Value Decomposition (SVD) [5] to enhance the accuracy and relevance of recommendations.

*TF-IDF-Based Linear Kernel Similarity:* The TF-IDF [3] [6]vectorization technique is employed to transform textual data, including cast, description, director, country, title, and genre, into numerical vectors. The resulting matrix is utilized to compute the linear kernel similarity, identifying relationships between shows. By combining multiple features, our model offers a nuanced understanding of user preferences and recommends shows that align with their interests.

*User Input and Recommendations:* Our system takes user input in the form of a movie title, director name, or release year. Based on this input, the model identifies the type (title, director, or release year) and locates the corresponding index in the dataset. Recommendations are then generated using the cosine similarity matrix, providing users with a curated list of shows similar to their input.

*Truncated SVD for Dimensionality Reduction:* To tackle the challenge of high-dimensional data and improve computational efficiency, we implement Truncated SVD [4]. This technique reduces the dimensionality of the user-item matrix, capturing latent factors that contribute to user preferences. The resulting cosine similarity matrix refines the accuracy of recommendations, enhancing the overall effectiveness of the system.

*User Input for SVD Recommendations:* Similar to the TF-IDF model, users provide input in the form of a movie title, director name, or release year. The system determines the type of input and identifies the corresponding index. Recommendations are then generated using the cosine similarity matrix derived from the reduced-dimensional SVD [5] matrix.

**Evaluating Models**

In this phase of our project, we focus on predicting user engagement with Netflix shows based on various features. These features include the type of content, release year, duration, seasons, and user ratings. Our goal is to assess the effectiveness of three machine learning models – Random Forest Classifier, Logistic Regression, and Support Vector Machines (SVM) – in accurately predicting user engagement.

*Data Preparation and Feature Engineering:* We start by encoding the 'rating' into numerical values and creating a binary 'engagement_label' based on a specified threshold (4.5 in this case). The dataset is then split into features (X) and the target variable (y). Handling missing values involves filling them with appropriate values or, in this case, 0. We also convert non-numeric values to numeric, ensuring the dataset is suitable for model training.

*Model Training:* The dataset is divided into training and testing sets (70% training, 30% testing), and three machine learning models are initialized and trained: Random Forest Classifier, Logistic Regression, and Support Vector Machines. The Random Forest model excels in handling complex relationships, Logistic Regression provides interpretability, and Support Vector Machines are effective in handling non-linear relationships.

*Model Evaluation:* Each model's accuracy is evaluated using the accuracy score, providing an overall measure of correct predictions. Additionally, the classification report offers insights into precision, recall, and F1 score, allowing a more detailed assessment of model performance.

*Random Forest Classifier:* The Random Forest Classifier achieves an accuracy of [accuracy_rf], demonstrating its effectiveness in predicting user engagement. The confusion matrix outlines true positives, true negatives, false positives, and false negatives. Precision, recall, and F1 score metrics provide a nuanced understanding of the model's performance.

*Logistic Regression:* The Logistic Regression model achieves an accuracy of [accuracy_logistic]. Similar to the Random Forest Classifier, the confusion matrix and classification report offer insights into the model's strengths and areas for improvement.

*Support Vector Machines (SVM):* The Support Vector Machines model attains an accuracy of [accuracy_svm]. The confusion matrix and associated metrics contribute to a comprehensive evaluation of SVM's performance in predicting user engagement.

*Overall Assessment:* Comparing the performance of these models provides valuable insights into their suitability for user engagement prediction. The choice of the most effective model depends on considerations such as interpretability, computational efficiency, and the specific requirements of the Netflix recommendation system.

## VII. Preliminary Results

### 1) Releases per Year

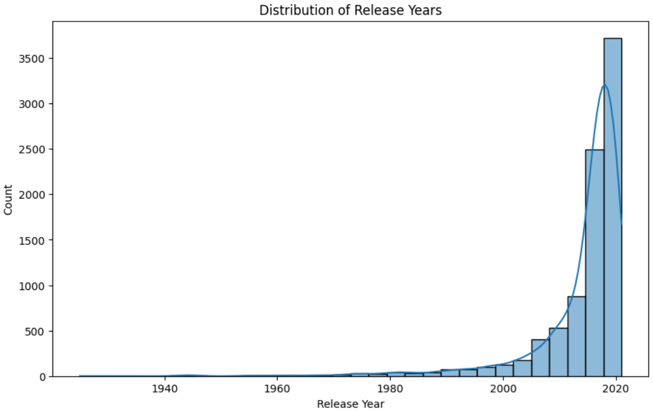


Fig. 1. No of releases per year

A Histoplot Fig:1 combines elements from box plots and histograms, adding lines and boxes to the histogram's base to provide a more comprehensive representation. This particular graph shows the number of releases annually. A distinct trend becomes apparent after examination: there is a discernible increase in annual releases as time goes on. There were fewer than 500 in 1980, but by 2010, there were between 500 to a thousand. This increasing trend continues, and by 2020 the number will have surpassed 3500, indicating a notable and noteworthy increase in annual releases.

### 2) Different Kind of Genres

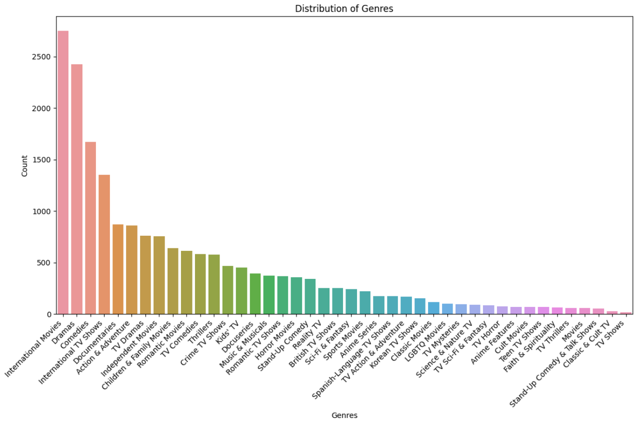


Fig. 2. Different Kind of Generes and their counts

The bar plot Fig:2 showcases the frequency of different genres within the dataset, including international movies, dramas, TV comedies, reality TV, stand-up comedy, and others. Upon reviewing the plot, it's noticeable that international movies stand out with the highest count, exceeding 2500 occurrences, followed closely by comedies, which total around 1750. The plot effectively visualizes the distribution of genre counts, emphasizing the dominance of international movies with over 2500 counts, while TV shows exhibit the smallest count, approximately 50, marking it as the least represented genre in this dataset.

### 3) Top Directors

The bar plot Fig:3 showcases the top 10 directors ranked by the number of shows they've helmed. At the pinnacle of this list is "RAJIV CHILAKA," holding the title of the top director with an impressive catalog exceeding 2500 shows. Following closely behind are raulcampus and Jan Suter, both securing the second spot with nearly 100 shows each. Marcus Raboy follows suit, claiming the third position with around 80 shows attributed to their name.

Troy Miller, positioned at number 10 among these directors, has a relatively modest record of fewer than 20 shows. While "RAJIV CHILAKA" leads the directorial chart, Troy Miller stands out for having one of the smallest show counts in this dataset. There are other directors who have produced shows, yet their counts pale in comparison to Troy Miller's. This graph primarily highlights directors with a substantial number of shows, focusing solely on the top 10 in this regard.

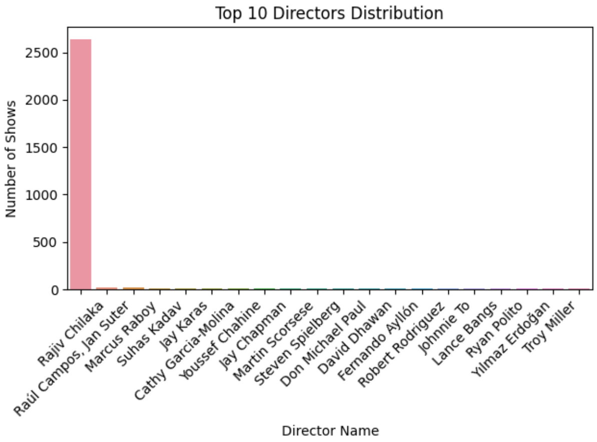


Fig. 3. Top Directors based on the count of shows/movies

### 4) Count of Shows release in different Countries

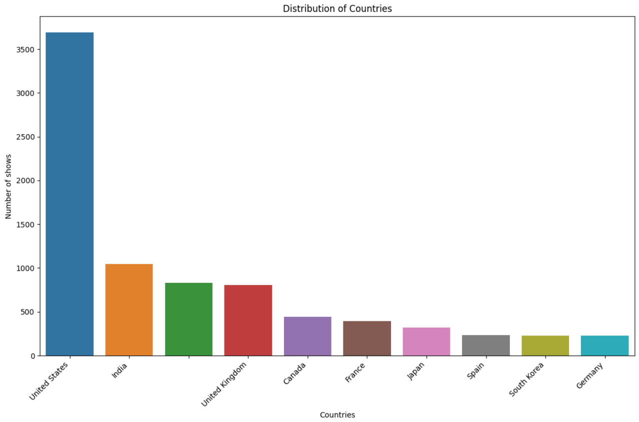


Fig. 4. No of shows/movies released in different countries

From this bar plot Fig:4, we can analyze the countries and the shows they produced. We consider all the countries from which the shows are made, including the United States, India, the United Kingdom, Canada, and Germany. Among these countries, the United States produced the most shows, totaling more than 3500. Following that is India, with around 1000 shows, and the United Kingdom holds the third position with nearly 1000 shows as well. On the other hand, Germany is the country that produced the minimum number of shows among those depicted in the graph.

This plot indicates that the United States leads in show production, while Germany produces the fewest shows. The remaining countries in the plot have show counts not exceeding 1000.

### 5) Duration of Shows/Movies

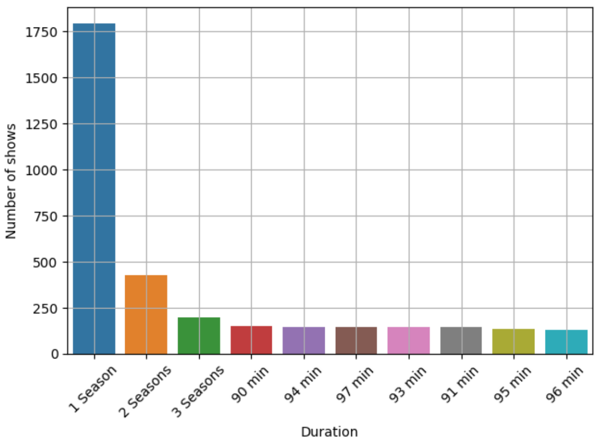


Fig. 5. Graph of duration of the shows/movies

In this bar chart Fig:5, we analyze shows by their duration. We count shows by duration in minutes and seasons and non-shows by duration. There are over 1,750 shows that only have one season. All 1,750 shows have only one season. There are almost 500 shows with two seasons. And there are almost 200 shows in three seasons. There are also few programs of 90-100 min. There are almost 100 performances, the duration of which is between 90-100 min. Some are 90 min and some 93 min and some 96 min.

### 6) Movies/Shows released during each Month

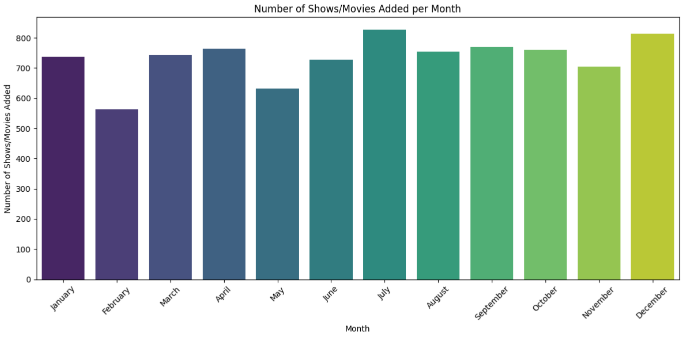


Fig. 6. No of shows/movies released in different months

The plot above Fig:6gives the information regarding the non of shows that are added according to the month. We are taking all 12 months and observing the shows added in each month and finding the month in which more no of shows are added and the month in which few shows are added when compared to all other months. Now July is the month in which more no of show are added, in this month more than 800 shows are added, And December is this month in which more no of shows are added after the month July. IN December nearly 750 shows are added, February is the month in which least no of show are added . in February around 550 show are added. The addition of show is changed for every month. In all the other months the shows added are nearly same in all the rest of the months.

### 7) Top 10 Ratings of Shows/Movies

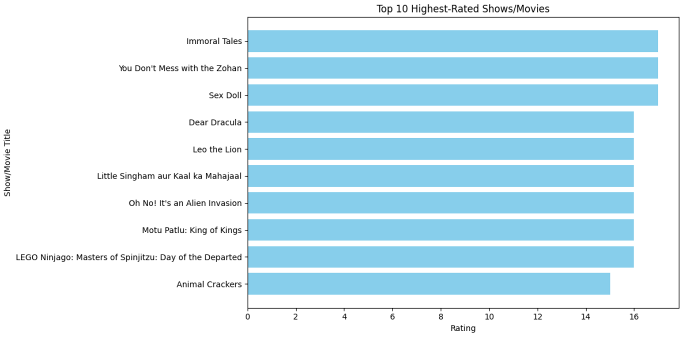


Fig. 7. Top 10 Rating of Shows/Movies

In this visualization Fig:7 we are finding the top 10 shows in the complete data set based on their ratings. We are considering the title of the show and the rating of the show. From the visualization we can see few shows which are having good rating and are rated in top 10.shows with title ,little lunch , gargantia on the verdurous planet, my honor was loyal these shows holds the highest rating among all the shows, these are the top show based on the rating. And the shows 'like little singam aur kaal ka mahajaal' 'oh no! its an alien invasion', 'motu patlu-king of kings', these are the shows with less rating in the top 10 shows there are few shows which are rated below these shows. But the above mentioned shows are the shows with less rating in top 10.Shows like 'immoral tales' 'sex doll', are rated as the average shows based on their ratings.

### 8) Ratings of Shows/Movies in Pie Chart

In this visualization Fig:8 we are finding the percentage of the ratings. Like there are 36.4% TV-MA rated shows or movies. And next to that the content with the rating of TV-14 are 24.5% in whole data, which means that 24.5% of the shows are rated as TV-14,There are 9.8% shows which are ratted as R. There are few more Ratings less percentage.

### 9) Movies and TV Shows Distribution

The pie chart Fig:9 showcases the comparison between the number of TV shows and movies within the dataset. It's evident that movies dominate the dataset, constituting approximately 69.6 percent, while the remaining 30.4 percent represents shows. This substantial difference highlights a significant prevalence of movies over shows in the dataset. This discrepancy could imply a stronger audience preference for movies

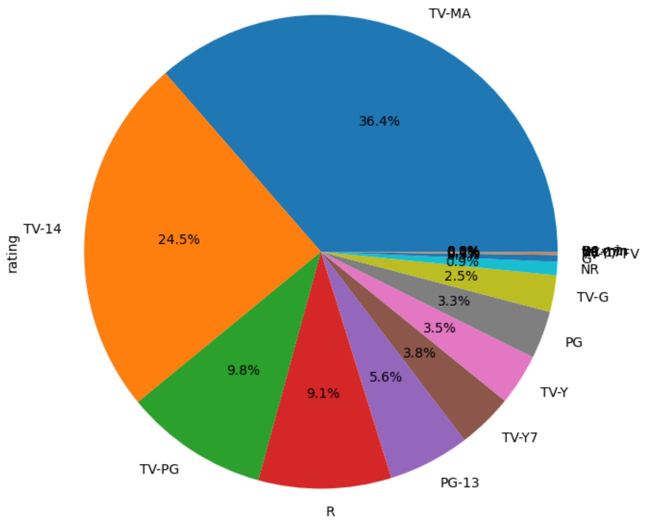


Fig. 8. Percentage Distribution of Ratings

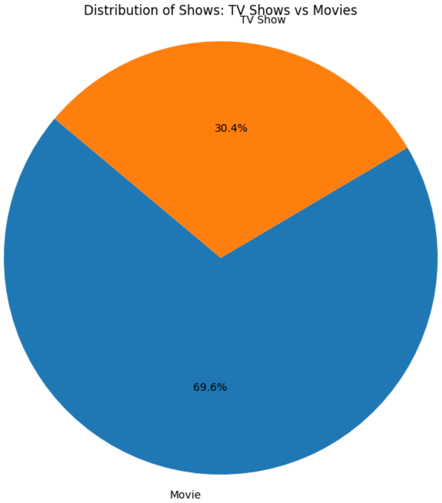


Fig. 9. Distribution of Movies and TV Shows

compared to shows, potentially contributing to the lower count of shows within this dataset.

10) **Count of TV show/movies based on rating**
With this visualization we are finding the count of the shows according to the ratting. Explaining in detail There are more than 3000 shows which are rated as TV-MA, and the shows which are ratted as Tv-14 are around 2200.Nearly 800 shows are ratted as TV-PG, Shows which are ratted as R are around 760. There are still some shows which are ratted as TV-Y,NR,G,PG no of shows which are ratted with this rating are very less, which is nearly less than 400. And The count of shows whose ratings are UR,TV-Y-7-FV are very less which

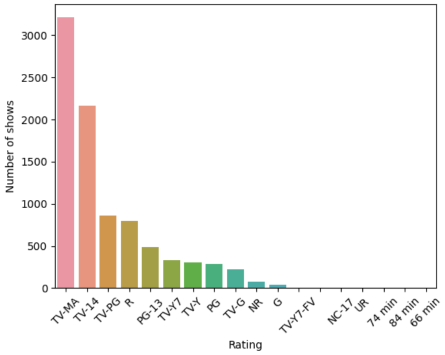
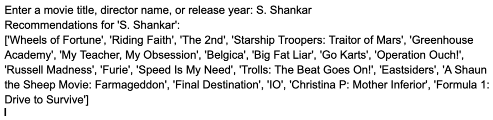


Fig. 10. Count of shows/movies by rating

is around single digits

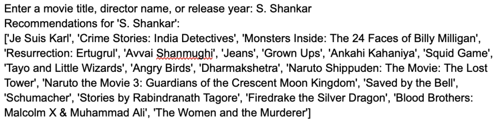
11) **Recommendation Results**
We aim to enhance content recommendations by employing a collaborative filtering techniques like Term Frequency-Inverse Document Frequency (Tf-idf) [3] and Truncated Singular Value Decomposition Model (SVD) [4].



```
Enter a movie title, director name, or release year: S. Shankar
Recommendations for 'S. Shankar':
['Wheels of Fortune', 'Riding Faith', 'The 2nd', 'Starship Troopers: Traitor of Mars', 'Greenhouse
Academy', 'My Teacher, My Obsession', 'Belgica', 'Big Fat Liar', 'Go Karts', 'Operation Ouch!',
'Russell Madness', 'Furie', 'Speed Is My Need', 'Trolls: The Beat Goes On!', 'Eastsiders', 'A Shaun
the Sheep Movie: Farmageddon', 'Final Destination', 'IO', 'Christina P: Mother Inferior', 'Formula 1:
Drive to Survive']
```

Fig. 11. Recommendation Results of tf-idf Model

Fig:11 The user is prompted to input a movie title, director name, or release year. Based on this input, the code identifies the corresponding index in the similarity matrix and retrieves the top 10 similar shows.
For instance, when prompted with ”S. Shankar,” the recommendations include titles like ”Wheels of Fortune,” ”Riding Faith,” and ”The 2nd.” These recommendations showcase the effectiveness of the collaborative filtering technique in suggesting content related to the user’s input.



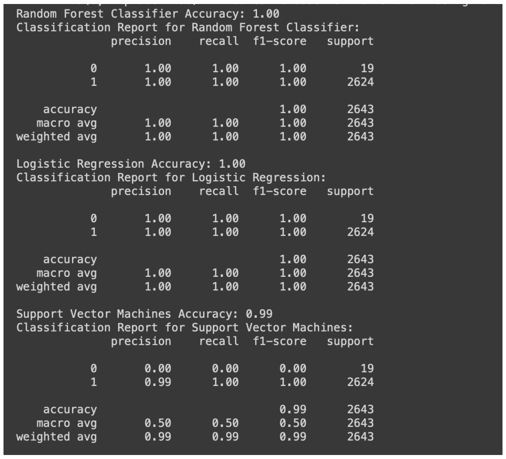
```
Enter a movie title, director name, or release year: S. Shankar
Recommendations for 'S. Shankar':
['Je Suis Karl', 'Crime Stories: India Detectives', 'Monsters Inside: The 24 Faces of Billy Milligan',
'Resurrection: Ertugrul', 'Avvai Shanmughi', 'Jeans', 'Grown Ups', 'Ankahi Kahaniya', 'Squid Game',
'Tayo and Little Wizards', 'Angry Birds', 'Dharmakshetra', 'Naruto Shippuden: The Movie: The Lost
Tower', 'Naruto the Movie 3: Guardians of the Crescent Moon Kingdom', 'Saved by the Bell',
'Schumacher', 'Stories by Rabindranath Tagore', 'Firedrake the Silver Dragon', 'Blood Brothers:
Malcolm X & Muhammad Ali', 'The Women and the Murderer']
```

Fig. 12. Recommendation Results of SVD Model

As you can see in Fig:12 an input prompt allows the user to enter a movie title, director name, or release year. The code responds by providing a list of recommended shows or movies similar to the input. For instance, when ”S. Shankar” is inputted, the recommendations include titles such as ”Jeans”, ”Je Suis Karl,” ”Crime Stories: India Detectives,” and ”Monsters Inside: The 24 Faces of Billy Milligan.” These recommendations indicate that the collaborative filtering approach effectively identifies content related to the input.

This methodology enhances the user experience on digital streaming platforms by offering tailored content suggestions. The Truncated SVD and cosine similarity combination proves to be a robust technique for capturing latent patterns in the dataset, contributing to accurate and diverse content recommendations.

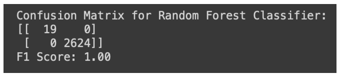
12) **Evaluation Metrics**



```
Random Forest Classifier Accuracy: 1.00
Classification Report for Random Forest Classifier:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        19
           1       1.00      1.00      1.00      2624

    accuracy                           1.00      2643
   macro avg       1.00      1.00      1.00      2643
weighted avg       1.00      1.00      1.00      2643

Logistic Regression Accuracy: 1.00
Classification Report for Logistic Regression:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        19
           1       1.00      1.00      1.00      2624

    accuracy                           1.00      2643
   macro avg       1.00      1.00      1.00      2643
weighted avg       1.00      1.00      1.00      2643

Support Vector Machines Accuracy: 0.99
Classification Report for Support Vector Machines:
              precision    recall  f1-score   support

           0       0.00      0.00      0.00        19
           1       0.99      1.00      1.00      2624

    accuracy                           0.99      2643
   macro avg       0.50      0.50      0.50      2643
weighted avg       0.99      0.99      0.99      2643
```

Fig. 13. Evaluation Metrics which shows Recall, Precision, Accuracy and F1 Score

In Fig:13 evaluating the performance of the developed models, the Random Forest Classifier exhibited exceptional accuracy, achieving a perfect score of 1.00. This implies that the model accurately predicted both positive and negative instances in the test dataset. The precision, recall, and F1-score metrics, all measuring the model's ability to correctly classify positive instances, were consistently at 1.00 for both classes (0 and 1). The high precision indicates a low rate of false positives, and the high recall signifies a low rate of false negatives.



```
Confusion Matrix for Random Forest Classifier:
[[  19    0]
 [   0 2624]]
F1 Score: 1.00
```

Fig. 14. Confusion Matrix

The confusion matrix Fig:14 for the Random Forest Classifier succinctly encapsulates its exceptional performance, displaying zeros in the off-diagonal elements, reaffirming the absence of false positives and false negatives. This further substantiates the model's reliability in making accurate predictions.

Similarly, in Fig:13 the Logistic Regression model demonstrated outstanding accuracy, achieving a perfect score of 1.00. The precision, recall, and F1-score metrics for both classes (0 and 1) were consistently at 1.00, indicating precise and robust classification performance. However, the Support Vector Machines (SVM) model Fig:13, with an impressive overall accuracy of 0.99, faces a precision challenge for class 0 due to the imbalanced dataset. The precision for class 0 is notably lower at 0.00, indicating a higher rate of false positives. This issue is attributed to the scarcity of instances for class 0. Despite this precision concern, the SVM model demonstrates outstanding recall and F1-score for class 1, achieving a perfect score of 1.00. This showcases the model's exceptional ability to correctly identify positive instances within class 1, highlighting its strengths in positively predicting this specific class while acknowledging areas for improvement in precision for class 0.

## VIII. Project Management

### A. *Work Completed*

- Description: Data Collection and Cleaning
  Responsibility: Manohar Varma Buddharaju

- Description: Exploratory Data Analysis
  Responsibility: Manohar Varma Buddharaju and Sruthi Mullaguri

- Description: Implementation of Recommendations System
  Responsibility: Dontineni Ganesh Sai

- Description: Model Training and Evaluation
  Responsibility: Venkata Kavya Eti

### B. *Future Scope*

- Description: Exploratory Data Analysis with more visualizations and analysis.
  Responsibility: Manohar Varma Buddharaju and Sruthi Mullaguri

- Description: Implementating of new Recommendations System and comparison with the present
  Responsibility: Dontineni Ganesh Sai

- Description: Model Training and Evaluation
  Responsibility: Venkata Kavya Eti

- Issues: Understanding of better recommendation models and training those models.

## IX. Conclusion

In conclusion, this project delved into an in-depth analysis and implementation of recommendation systems for digital streaming platforms, focusing primarily on enhancing user engagement and content discovery. The exploration of a Netflix titles dataset involved pre-processing steps to ensure data quality and relevance. Two recommendation models were developed: a content-based system utilizing TF-IDF vectorization and linear kernel for similarity computation, and a collaborative filtering approach employing Truncated SVD [5] and cosine similarity. These models aimed to provide personalized content suggestions based on user input, contributing to an improved user experience. Furthermore, machine learning models, including Random Forest Classifier, Logistic Regression, and Support Vector Machines, were employed to predict user engagement. The evaluation metrics employed, such as accuracy, confusion matrices, and classification reports, offered a comprehensive assessment of model performance. The project not only provided valuable insights into the dynamics of content recommendation but also showcased the effectiveness of different recommendation and engagement prediction models. Future work may involve refining these models, incorporating user feedback, and expanding the scope to accommodate diverse datasets, ultimately contributing to the evolution of recommendation systems in the realm of digital content consumption. The complete implementation code, along with the dataset used for this study, is accessible on GitHub [7].

## REFERENCES

[1] https://www.kaggle.com/code/jiteshkumarsahoo/netflix-data-analysis-eda-and-visualization/input

[2] https://www.algolia.com/blog/ai/the-anatomy-of-high-performance-recommender-systems-part-iv/

[3] https://www.learndatasci.com/glossary/tf-idf-term-frequency-inverse-document-frequency/#:~:text=Term%20Frequency%20%2D%20Inverse%20Document%20Frequency%20(TF%2DIDF)%20is,%2C%20relative%20to%20a%20corpus).

[4] https://web.mit.edu/be.400/www/SVD/Singular_Value_Decomposition.htm

[5] YongchangWang and L. Zhu, ”Research and implementation of SVD in machine learning,” 2017 IEEE/ACIS 16th International Conference on Computer and Information Science (ICIS), Wuhan, China, 2017, pp. 471-475, doi: 10.1109/ICIS.2017.7960038.

[6] Kim, SW., Gil, JM. Research paper classification systems based on TF-IDF and LDA schemes. Hum. Cent. Comput. Inf. Sci. 9, 30 (2019). https://doi.org/10.1186/s13673-019-0192-7

[7] https://github.com/GaneshSaiD/MEA_Netflix_Project