**Netflix Movies and Tv Shows Clustering**

**Puneet Suthar**

**Data Science Enthusiast**

**AlmaBetter, Bangalore**

**Abstract:**

Netflix operates as a streaming service, offering an extensive library of movies and TV shows accessible online at any time. The company derives its revenue from users who make monthly payments to utilize the platform, but customers can cancel their subscriptions without restrictions. Consequently, Netflix must ensure that users remain engaged with the platform and not lose their interest. To achieve this, recommendation systems play a crucial role by providing users with valuable suggestions to enhance their viewing experience.

1. **Introduction**

Netflix's recommendation system is instrumental in boosting its appeal among service providers. It facilitates an increase in sales, provides a diverse range of options, enhances user satisfaction, strengthens customer loyalty, and helps to understand user preferences better. This, in turn, makes it easier for users to make informed decisions from a vast selection of movies. With 139 million paid subscribers, 15,400 regional library titles, and 112 Emmy Award nominations in 2018, Netflix reigns supreme as the world's top internet television network and largest streaming service. The company's remarkable digital success story is incomplete without acknowledging the pivotal role of its recommender systems, which prioritize personalization. To generate a list of recommendations tailored to individual preferences, Netflix utilizes methods such as Collaborative Filtering and Content-based Filtering.

1. **Problem Statement**

This dataset consists of tv shows and movies available on Netflix as of 2019. The dataset is collected from Flixable which is a third-party Netflix search engine.

In 2018, they released an interesting report which shows that the number of TV shows on Netflix has nearly tripled since 2010. The streaming service’s number of movies has decreased by more than 2,000 titles since 2010, while its number of TV shows has nearly tripled.

**In this project, you are required to do**

1. Exploratory Data Analysis
2. Understanding what type content is available in different countries
3. Is Netflix increasingly focused on TV rather than movies in recent years?
4. Clustering similar content by matching text-based features.
5. **Dataset Description**

The dataset provided contains 7787 rows and 12 columns.

The following are the columns in the dataset:

1. **Show id:** Unique identifier of the record in the dataset
2. **Type**: Whether it is a TV show or movie
3. **Title:** Title of the show or movie
4. **Director:** Director of the TV show or movie
5. **Cast:** The cast of the movie or TV show
6. **Country:** The list of the country in which a show/ movie is released or watched
7. **Date added:** The date on which the content was onboarded on the Netflix platform
8. **Release year:** Year of the release of the show/ movie
9. **Rating:** The rating informs about the suitability of the content for a specific age group
10. **Duration:** Duration is specified in terms of minutes for movies and in terms of the number of seasons in the case of TV shows
11. **Listed in:** This columns species the category/ genre of the content
12. **Description:** A short summary about the storyline of the content
13. **Steps involved**
    1. Performing EDA (exploratory data analysis).
    2. Clustering of data.
    3. Movie recommendation Using Cosine Simillarity and defined cluster for filtering
14. **Performing EDA (exploratory data analysis)**
15. Exploring head and tail of the data to get insights on the given data.
16. Looking for null values, No. of zeros, duplicates, no. of unique in every column, it help us to make a guideline for feature engineering section before major EDA.
17. We have 2389 null values in director column. We have almost 30% null values in this column. so replacing null with unknown.
18. We have 718 null values in cast column. and it can be replaced with 'unknown'.
19. We have 507 null values in country column. Replacing nulls with 'mode' value that is USA.
20. Also we have 10 null values in date\_added column. we have few rows of date\_added so we can 'drop' these rows.
21. As rating column has 0.08% null values , so replacing nulls with most frequent TV-MA rating.
22. We left with 7777 Rows.
23. We created new colums named day\_added, month\_added, year\_added.
24. Adding new features based the list of above converted lists:-

1. number\_of\_director

2. number\_of\_cast

3. number\_of\_countries

4. number\_of\_genres

1. In Major EDA we Found:

**1. 'type' column**

* According to the graph we have 5377(69.1%) movies,
* And 2400(30.86%) as TV Show in this dataset.

**2. 'director' column**

* Top 3 directors are:-
  1. Jan Suter
  2. Raul Campos
  3. Marcus Raboy

**3. 'cast' column**

* Top 4 Casts are:-
  1. Anupam Kher - 45 listings
  2. Shah Rukh Khan - 35 listings
  3. Naseeruddin Shah - 30 listings
  4. Om Puri - 30 listings
  5. TOP CAST COMBINATIONS:
     1. David Attenborough: 10 listings
     2. Jeff Dunham: 7 listings
     3. Louis C.K.: 5 listings
     4. Jim Gaffigan: 4 listings

**4. 'country' column**

* Top 3 Countries are:-
  1. United States - 3797 listings - 39.7%
  2. India - 990 listings - 10.4%
  3. United Kingdom - 722 listings - 7.5%

**5. 'date\_added' column:-**

* The data shows a significant increase in the number of releases from 2015 to 2020.
* During this period, there were notable spikes in the number of releases in the months of October (785), November (738), December (833), and January (757), which could be due to the holiday season.
* However, there was a sudden drop in 2021, which could be attributed to the impact of the COVID-19 pandemic.
* Most of the content gets uploaded in the beginning and the middle of the month.

**6. 'release\_year' column**

* We have 744 movies and 268 TV Show releases in 2017
* Also 734 movies and 386 TV Show releases in 2018
* Most of the content was released between 2010 and 2021

**7. 'Rating' column**

* As we can see the distribution is not correct, because some movies are rated in TV rating and vice versa. so need to be fixed
* Most number of movies rated R i.e. Adult Rating
* Most number of TV Shows rated TV-MA i.e. Adult Rating

**8. 'duration' column**

* TV shows - The most frequently listed duration is season 1, with 1608 listings, followed by season 2 with 378 listings.
* Movies - The movie durations mainly range from 55 to 150 minutes, with the majority falling between 90 to 120 minutes.

**9. 'genre' column**

* In Movies, International\_movies is the most popular genre on Netflix.
* Drama is the second most popular genre on Netflix.
* Comedy is the third most popular genre on Netflix.
* Ranking is the same for TV shows.

**10. 'title' column**

* Most repeated words in title column are love, Christmas, World, Man, and life.

**11. 'description' column**

* Most repeated words in the description of the TV shows and movies are Family, new, Love, Life, mother, find.

1. **Clustering of data**

As the dataframe contain both type of data text and numeric, so it not good to label clusters only based on text data or numeric data, because different data type can give different clustering.

so we applied clustering algorithm on 3 sections of data set:-

**1. Grouping 1 :**

By applying clustering on the columns 'director', 'cast', 'country', 'listed\_in', we are trying to find similar shows based on their crew and category information. This approach can be useful to discover patterns in the production crew and show categories that can be used for content recommendations or to identify production trends in the entertainment industry.

To perform the above task Following feature Engineering has been done:-

* 1. One hot encoding of 'director', 'cast', 'country', 'listed\_in' columns in different dataframes
  2. But if we merge them all together in a single dataframe, then no. of columns are very high (37,486) columns
  3. To reduce no. of column we selected those cast, directors, which have minimum 5 listings then total downs to (1978) columns
  4. Then merged all OHE dataframe into one to apply kmeans.

\* Clustering results Using Silhoutte score, Elbow graph, Calinski Harabaz score we found optimum K :-

1. kmeans Cluatering : k = 7

2. Agglomerative clustering : k = 7

**kMeans performed well in group 1**

**2. Grouping 2:**

By using clustering on the columns ['type', 'release\_year', 'rating', 'country\_count',

'number\_of\_directors', 'number\_of\_casts', 'number\_of\_countries',

'number\_of\_genres'] we are trying to find similar shows based on more quantitative data such as show type, date added, ratings, etc. This approach can be useful for creating a recommendation system that is based on these attributes.

To perform the above task Following feature Engineering has been done:-

1. Applied minmaxscaler on release year because it was too high no. than others

Clustering results Using Silhoutte score, Elbow graph, Calinski Harabaz score we found optimum K :-

1. kmeans Cluatering : k = 7

2. Agglomerative clustering : k = 5

**kMeans performed well in group 2**

**3. Grouping 3:**

By applying LDA (Latent Dirichlet Allocation) on the 'description' column, we are trying to extract topics or themes present in the show descriptions and then using k-means clustering to group similar shows based on these themes. This approach can be useful to understand the content and genre of shows, and to discover patterns in the storytelling, themes and genre of shows.

We also applied kmeans on the infered vector of description column using doc2vec model, but it dosen't give good and clean clusters, thats why we used LDA.

To perform the above task Following feature Engineering has been done:-

* 1. To Use text data of description column we did some text preprocessing :-
* lower casing
* Remove punctuation
* Tokenize the text
* Removing Stop words
* Stemming using porter stemmer
* lemmatization using word net lemitization
* And join all of them in a string
  1. Then we calulated the coherence score using LDA multicore and coherence model from gensim to find the best no. of topics = 16
  2. Then we vectorized the column using TF-IDF
  3. Then applied LDA on the vectorized data

Clustering results :- Using Silhoutte score and Elbow graph we found optimum K = 16.

**3. Grouping 4:**

Applying clustering to all the data used in above clusters.

Clustering results Using Silhoutte score, Elbow graph, Calinski Harabaz score we found optimum K :-

1. kmeans Cluatering : k = 6

2. Agglomerative clustering : k = 6

**KMeans performed well in group 4**

We can say that, the different groupings of features used in k-means and agglomerative clustering can uncover different patterns and insights in the Netflix data frame, and the choice of features will depend on the specific use case and the questions being asked.

* 1. **Movie recommendation system**

This function find\_similar\_texts takes as input a dataframe of Netflix movie descriptions and titles, a pre-trained Doc2Vec model, a title of a movie to find similar texts to, and several optional parameters such as the number of similar texts to return, the column to use for clustering, and whether or not to filter the search by cluster. The function first infers the vector representation of the input text, and then filters the dataframe to only include texts with the same cluster label if specified. It then uses the most\_similar method from the model's Docvecs object to find similar texts, and returns the most similar texts along with their cluster label, similarity score, title, and description. The results are also printed in a readable format for easy visualization.

* 1. **Conclusion**

In conclusion, the Netflix dataset is a rich resource for various types of analyses, such as exploratory data analysis and clustering. Through our exploratory data analysis, we gained insights into the distribution of the dataset across various features such as type, directors, cast, country, date added, release year, rating, duration, genre, title, and description. We also carried out clustering on the dataset and discovered that there are several ways to group the shows based on their features. This can be useful in creating a recommendation system for Netflix viewers. The analysis revealed interesting patterns and trends in the Netflix dataset, which can be used to enhance the user experience on the platform.

**References**

1. Towards Data Science
2. Stack overflow
3. Medium