Capstone Project

Netflix Movie and TV Show Clustering

(Clustering, Content Based Recommendation System)

By - Jayesh Dahiwale

STEPS INVOLVED :

Steps that significantly contribute towards achieving the results:

1. Defining Problem Statement
2. Understanding Data
3. Exploratory Data Analysis
4. Data Cleaning
5. Textual Data Preprocessing
6. Clusters Implementation
7. Build Recommendation System
8. Conclusions

Project Goal :

The goal of this project is to classify/group the Netflix shows into certain cluster such that movie and TV shows that are in the same cluster/group should have similar properties and/or features, while data points in different groups should have highly dissimilar properties and/or features.

Problem Statement :

- Netflix is the world's largest online streaming service provider, with over 230 million subscribers as of 2023-Q1. It is crucial that they effectively cluster which shows that are hosted on their platform in order to enhance the user experience, thereby preventing subscriber churn.
- We will be able to understand the shows that are similar to each other and different from one another by creating clusters, which may be leveraged to offer the consumers personalized shows suggestions depending on their preferences.

DATA PIPELINE :

1. **Analyze Data:** In this initial step we went to look for different features available and tried to understand the data. During this stage, we looked for the shape of data,data types of each feature, statistical summary etc.

1. **EDA:** EDA or Exploratory Data Analysis is the critical process of performing initial investigation on the data. So,through this we have observed certain trends and dependencies from the dataset that will be useful for further processing.

1. **Data Cleaning:** Checked duplicated values present in the dataset. After that comes the null value and outlier detection and treatment. For the null values imputation we simply replace the empty string and drop some of the null rows then analyze outlier and handling.

1. **Textual Data Preprocessing:** During this stage, cluster the data based on the attributes: director, cast, country,genre, rating and description. Data preprocessing include Remove all stop words and punctuation marks, convert all textual data to lowercase. Stemming to generate a meaningful word out of corpus of words. Tokenization of corpus and Word vectorization. We used Principal Component Analysis(PCA) to handle the curse of dimensionality.

1. **Clusters Implementation:** Used K-Means and Agglomerative Hierarchical clustering algorithms to cluster the movies, obtained the optimal number of clusters using different techniques.

1. **Build Content Based Recommendation System:** A content-based recommender system was build using the similarity matrix obtained after using cosine similarity. This recommender system will display 10 recommendations to the user based on the type of movies/show they watched.

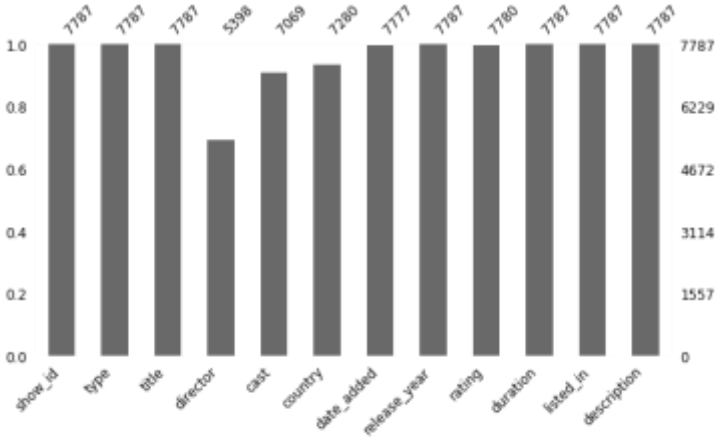
DATASET SUMMARY:

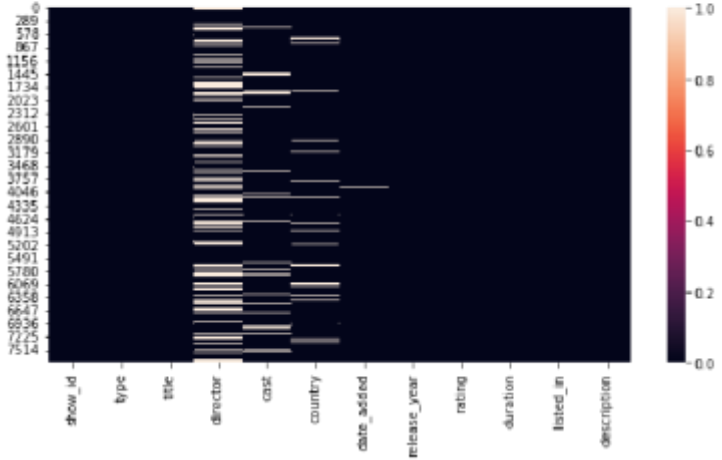
- This dataset consist of movies and TV shows available on Netflix till 2020.
- The dataset contained about 7787 records and 12 attributes.
- Most of the features are Present in textual format.
- Dataset contains two type of content one is Movie and another one is TV show.

ATTRIBUTE INFORMATION:

- **Show Id:**Unique Id for every Movie/Show
- **Type :** Identifier - Movie/Show
- **Title :** Title of the Movie/Show
- **Director:** Director of the Movie/Show
- **Cast:** Actors involved in the Movie/Show
- **Country:** Country where the Movie/Show was produced
- **Date Added:** date it was added on Netflix
- **Release Year:** Actual Release year of the Movie/Show
- **Rating:** Total Duration - in minutes or number of seasons
- **Listed In :** Genre
- **Description :** The summary description

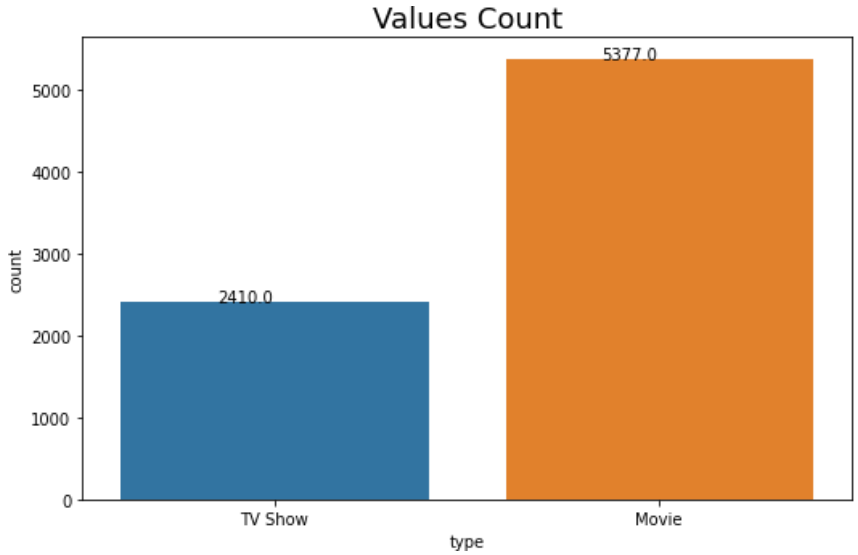
Null Values Treatment:

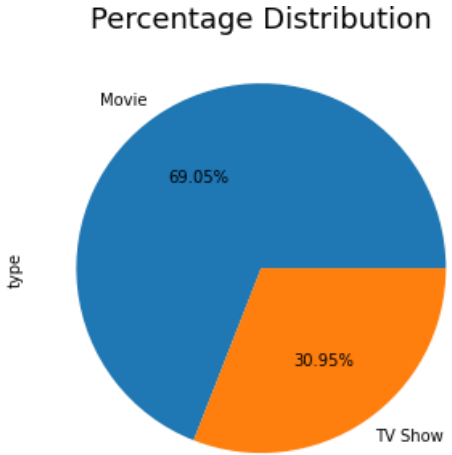




- Null values present in the director, cast, country, date_added and rating column.
- All the data that we have is related to each specific movie. So, we can't impute any null values with using any method. Also, we don't want to lose any data since the data size is small for that reason.
- The null values in the director, cast, and country attributes can be replaced with 'empty string’.
- Small amount of null value percentage present in rating and date_added column, if we drop these nan values it will not affect that much while building the model. So, we simply drop the nan value present in rating and date_added columns.

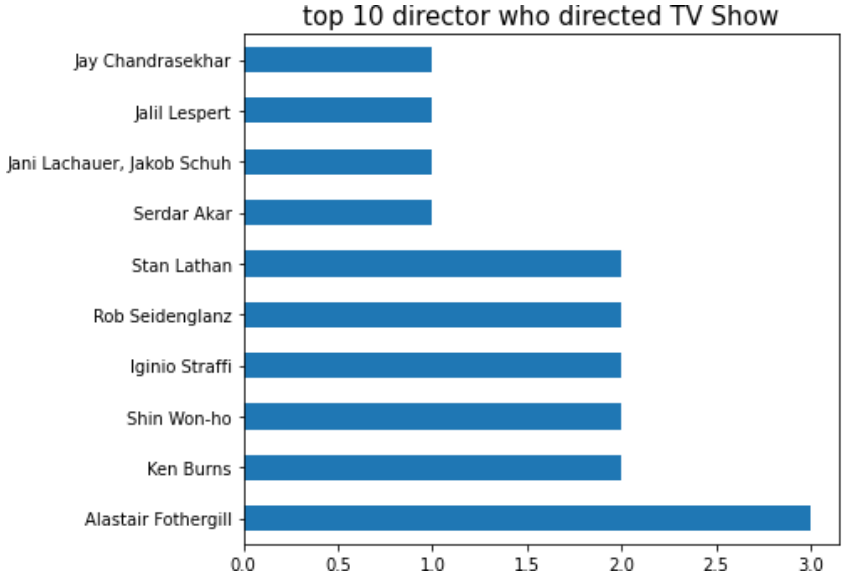
EXPLORATORY DATA ANALYSIS: (Type Column)

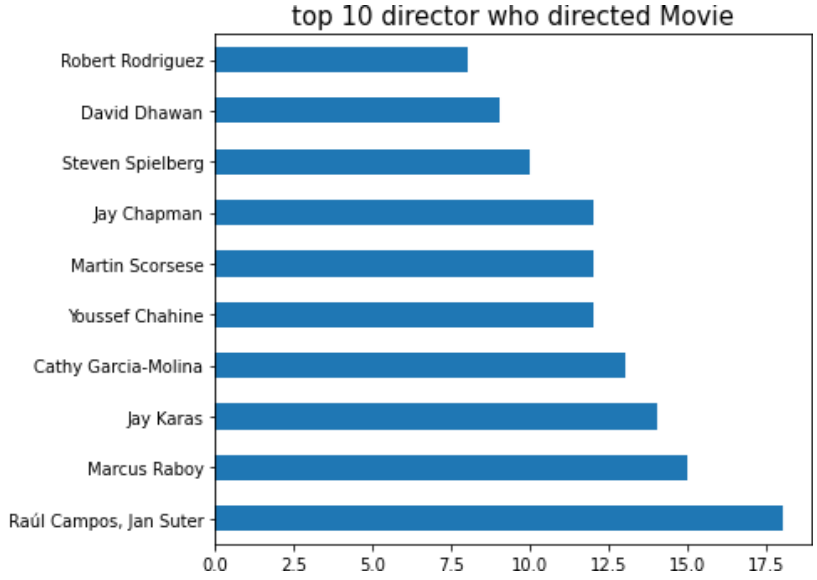




- There are a greater number of count is Movie than TV Show.
- 69% of data belong from Movie class and 31% of data belong from TV shows

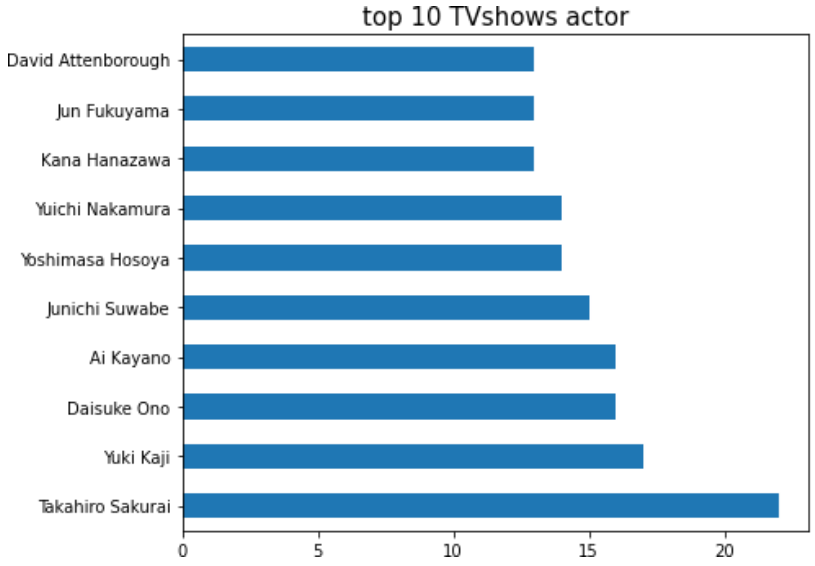
EXPLORATORY DATA ANALYSIS: (Director column)

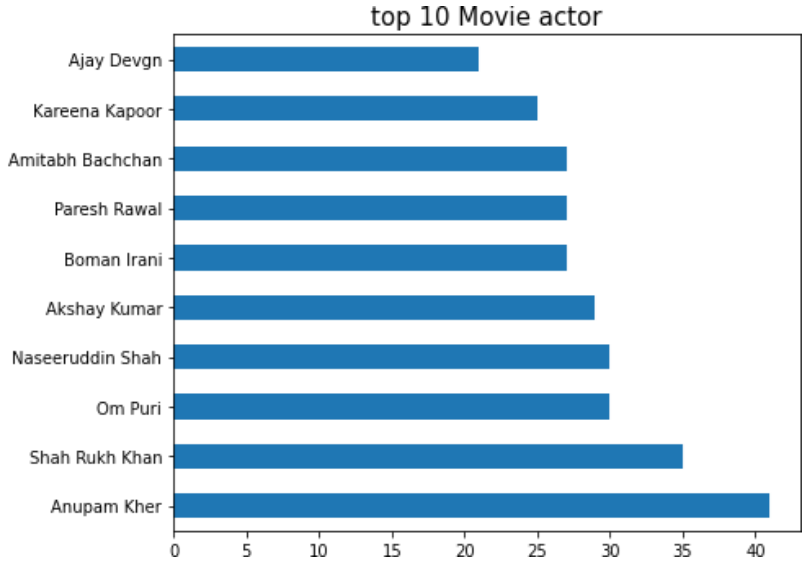




- Alastair Fothergill directed highest shows in data list which is 3.
- Raul Campos and Jan Suter together have directed 18 movies, higher than anyone in the dataset.

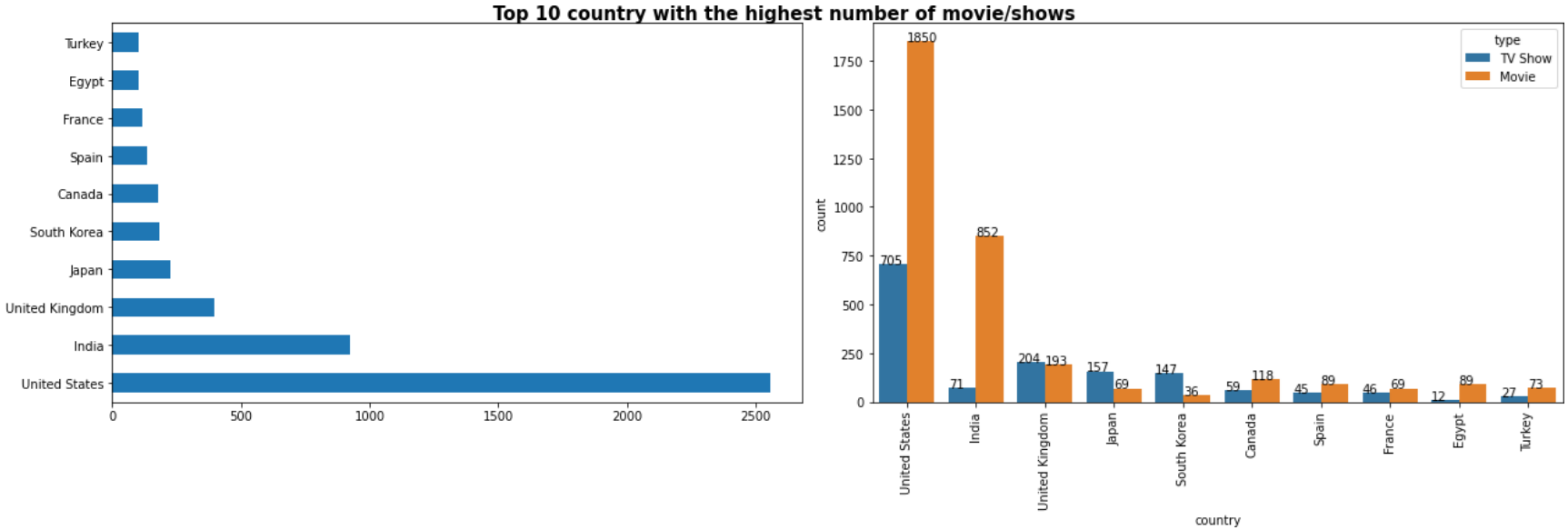
EXPLORATORY DATA ANALYSIS: (Cast column)





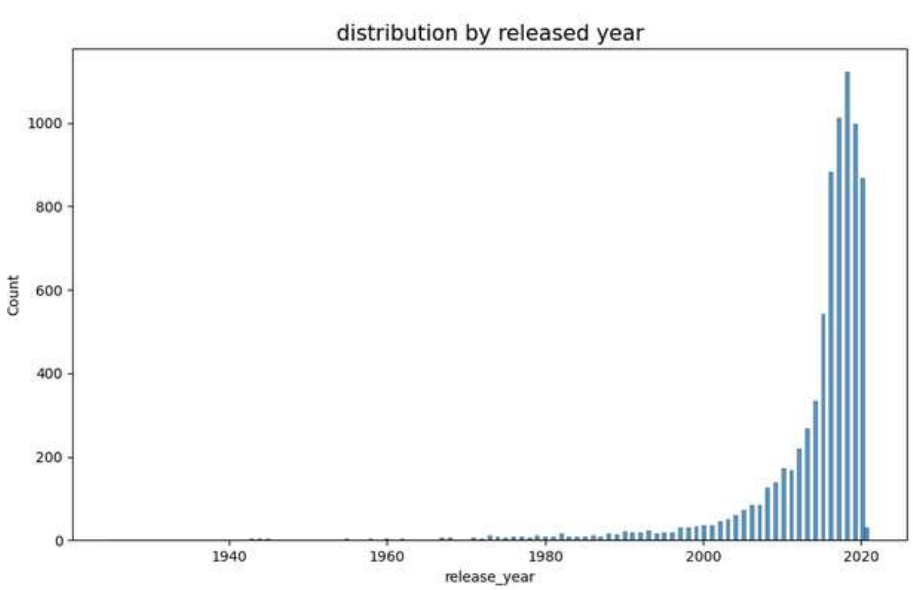
- Anupam Kher, Shahrukh Khan, Om Puri play highest number of role in the movies.
- Takahiro Sakurai, Yuki Kaji, Daisuke Ono play highest role in the TV shows

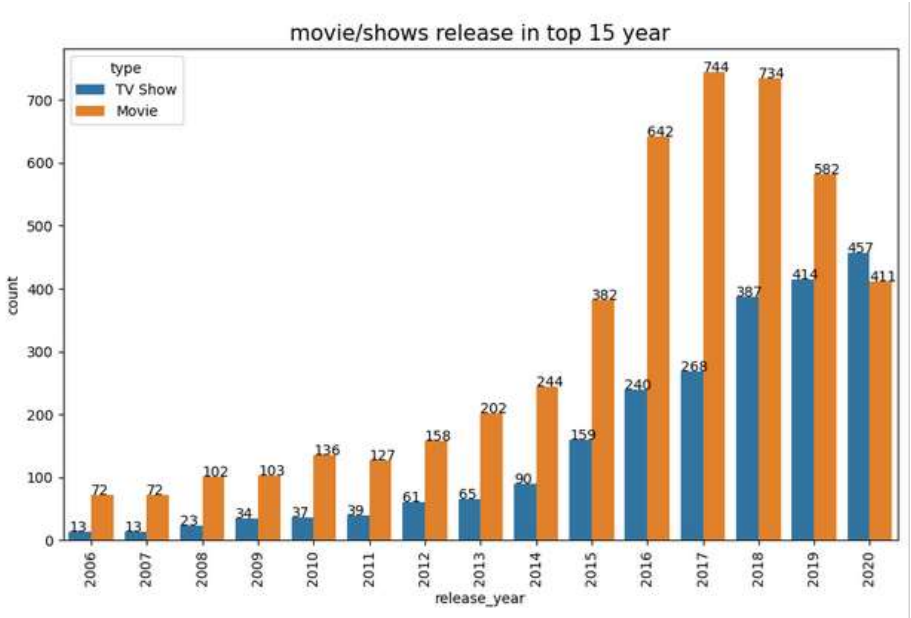
EXPLORATORY DATA ANALYSIS: (Country column)



- The highest number of count were based out of the United State, followed by India and United Kingdom.
- In India and United State, a greater number of movie present compared to TV show.
- Greater number of count in TV shows belong from South Korea and Japan.

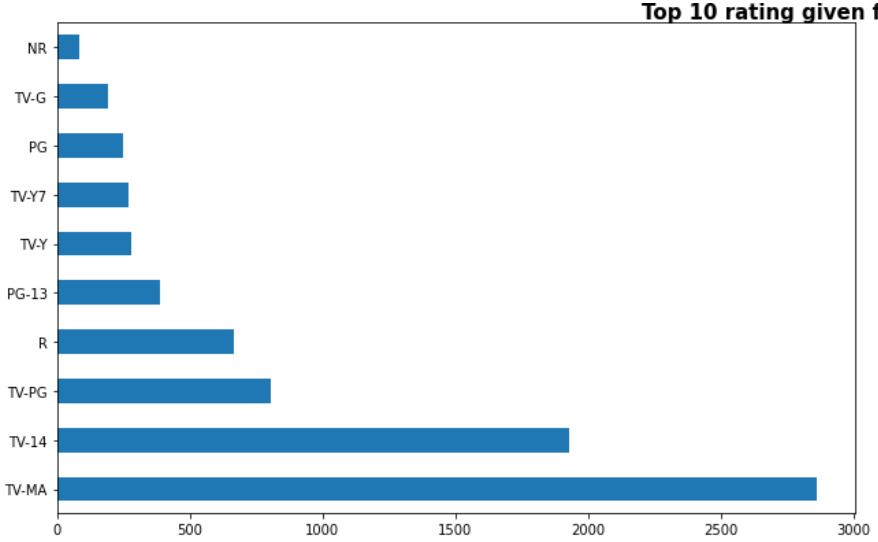
EXPLORATORY DATA ANALYSIS: (Release_Year column)

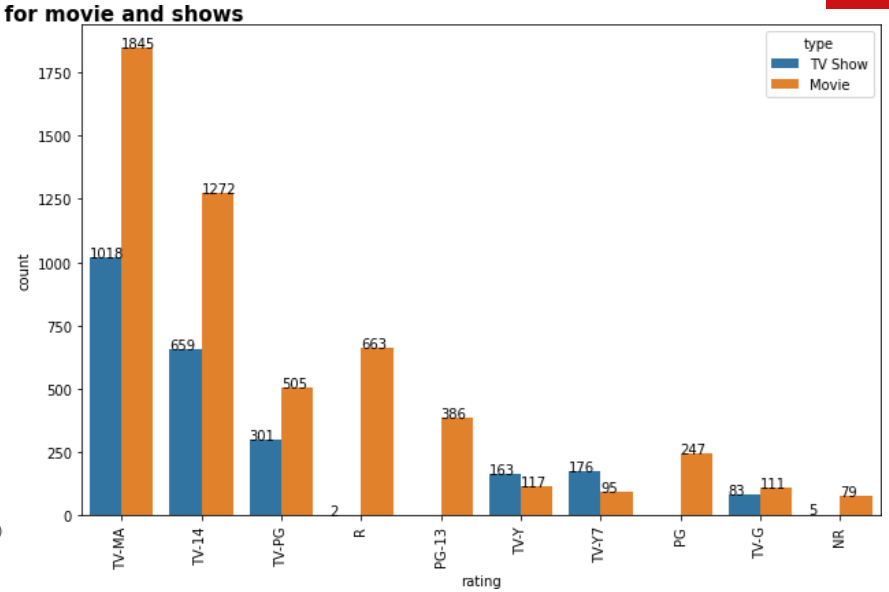




- Netflix has greater number of new movies / TV shows than the old ones.
- Highest number of movie/shows are released in Netflix in between 2015-2020 and highest number of count belong from 2018 year.

EXPLORATORY DATA ANALYSIS: (Rating column)

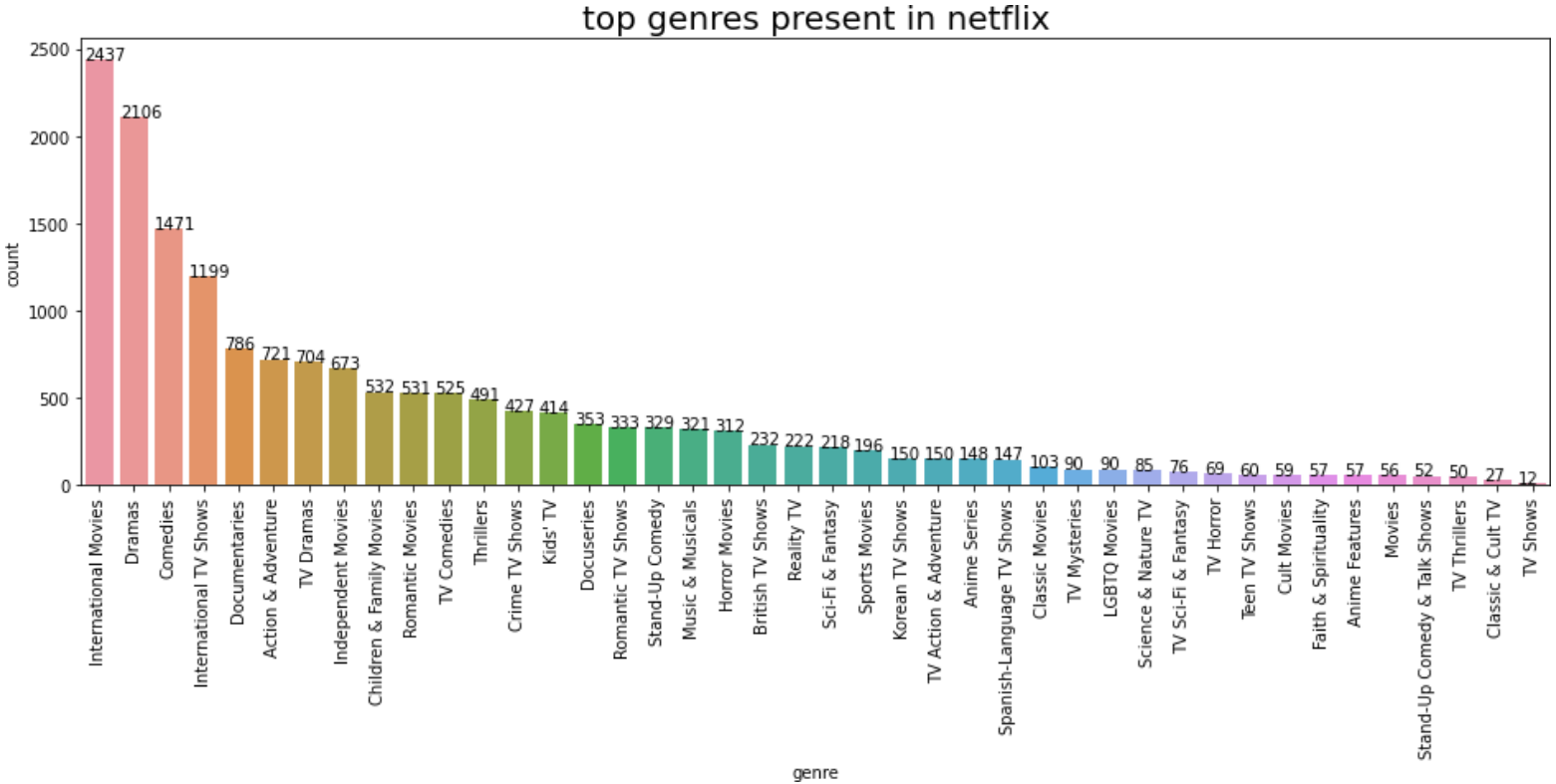




- TV-MA : Mature Adults
- R : Adults
- PG-13 : Teens
- TV-14 : Young Adults
- TV-PG : Older Kids
- NR : Adults
- TV-G : Kids
- TV-Y : Kids
- TV-Y7 : Older Kids

- Most of the Movies and tv shows have rating of TV-MA (Mature Audience) then followed by TV-14 (younger audience).
- Highest number of rating given for the movies it is obvious because of highest number of category belong from Movie class as we can seen in type column.

EXPLORATORY DATA ANALYSIS: (Listed_IN [genre] column)



- Highest number of genre belong from International movies, Dramas, Comedies, respectively.
- Least number of genre belong from Classic & Cult TV, TV Thriller, Stand-Up comedy and Talk show

Modeling Approach:

1. Select the attributes based on which you want to cluster the shows
2. Text preprocessing: Remove all stop words and punctuation marks, convert all textual data to lowercase.
3. Stemming to generate a meaningful word out of corpus of words.
4. Tokenization of corpus and Word vectorization
5. Dimensionality reduction
6. Use different algorithms to cluster the movies, obtain the optimal number of clusters using different techniques
7. Build optimal number of clusters and visualize the contents of each cluster using word clouds.

Cluster:

We create one cluster column based on the following features:

- Director
- Cast
- Country
- Rating
- Listed in (genres)
- Description

Before clusters implementation we need to pre-process the data. So that we filtered data with following steps:

AI

1. Removing Stop words

- Stop words are common words like “the”, “and” and “but” do not carry much meaning on their own and are often seen as noise in the data.

2. Lowercasing words

- Lowercasing the words can also reduce the size of the vocabulary, which can make it easier to work with larger texts or texts in languages with a high number of inflected forms.

3. Removing Punctuation

- Punctuation marks like periods, commas, and exclamation points can add noise to the data and can sometimes be treated as separate tokens, which can affect the performance of NLP models.

4. Stemming

- used Snowball Stemmer to generate a meaningful word out of corpus of words.
- For example, the words "run," "runs," "ran," and "running" are all different inflected forms of the same word "run," and a stemmer can reduce them all to the base form "run."

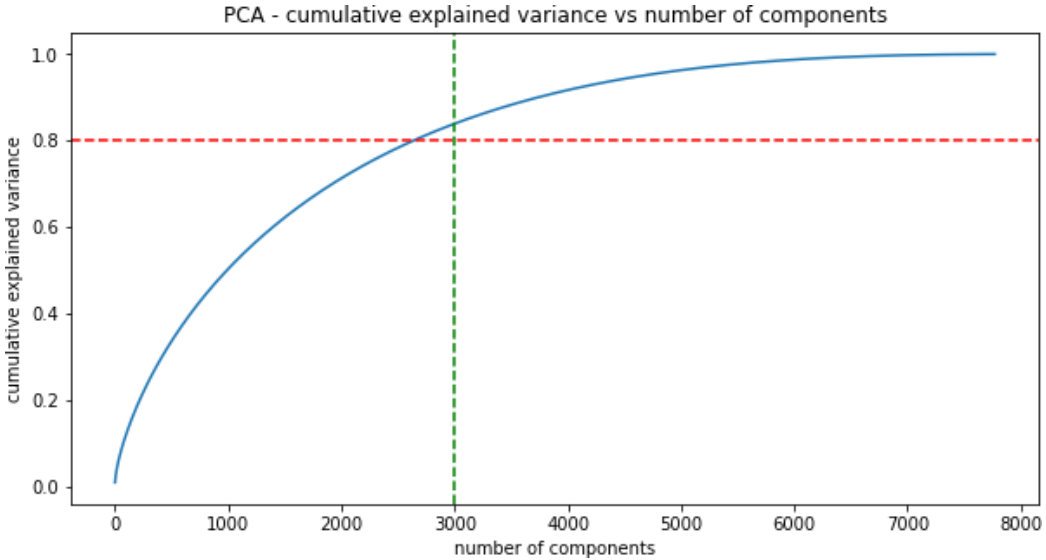
5. Tokenization of corpus and Word vectorization – TFIDF

1. This is important in NLP tasks because most machine learning models expect numerical input and cannot work with raw text data directly. Word vectorization allows you to input the words into a machine learning model in a way that preserves the meaning and context of the words.

6. Dimensionality reduction – PCA

- Dimensionality reduction is the process of reducing the number of features or dimensions in a dataset while preserving as much information as possible. As high-dimensional datasets can be difficult to work with and can sometimes suffer from the curse of dimensionality.

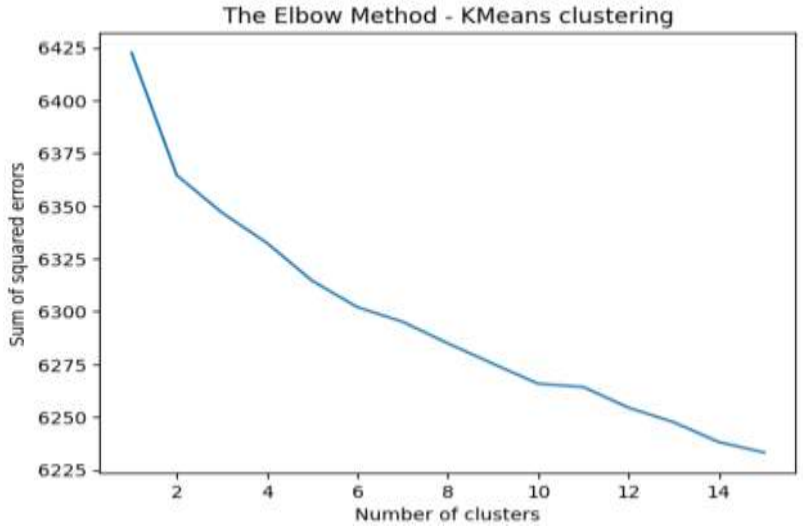
Principal Component Analysis:

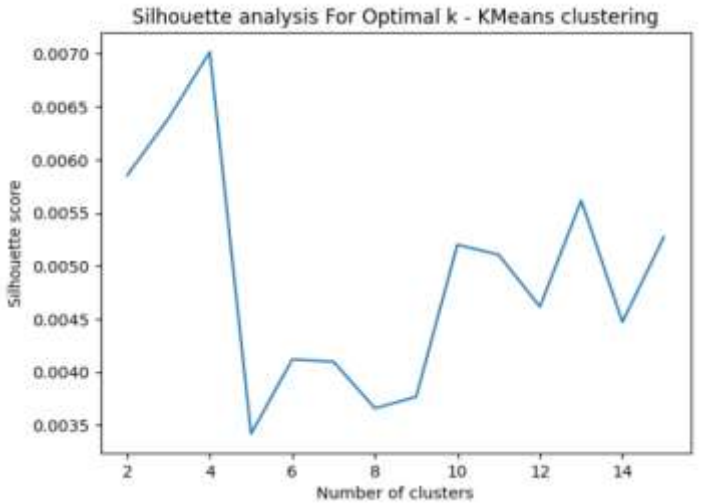


- As you can see that 100% of the variance is explained by about ~7500 components.
- Also, more than 80% of the variance is explained just by 3000 components.
- Hence to simplify the model, and reduce dimensionality, we take top 3000 components, which will still be able to capture more than 80% of variance

K-Means Clustering:

- Visualizing the elbow curve and Silhouette score to decide on the optimal number of clusters for K- means clustering algorithm.

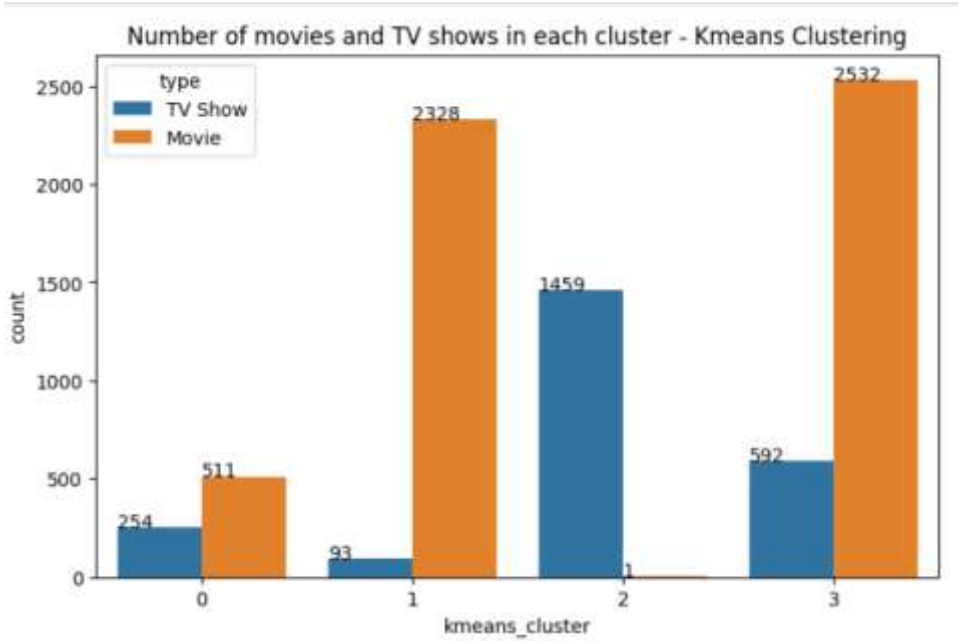




- The sum of squared errors between each point and the centroid in a cluster decreases with the increase in the number of clusters

- The highest Silhouette score is obtained for 4 clusters.
- Building 4 clusters using the k-means clustering algorithm

K-Means clusters:



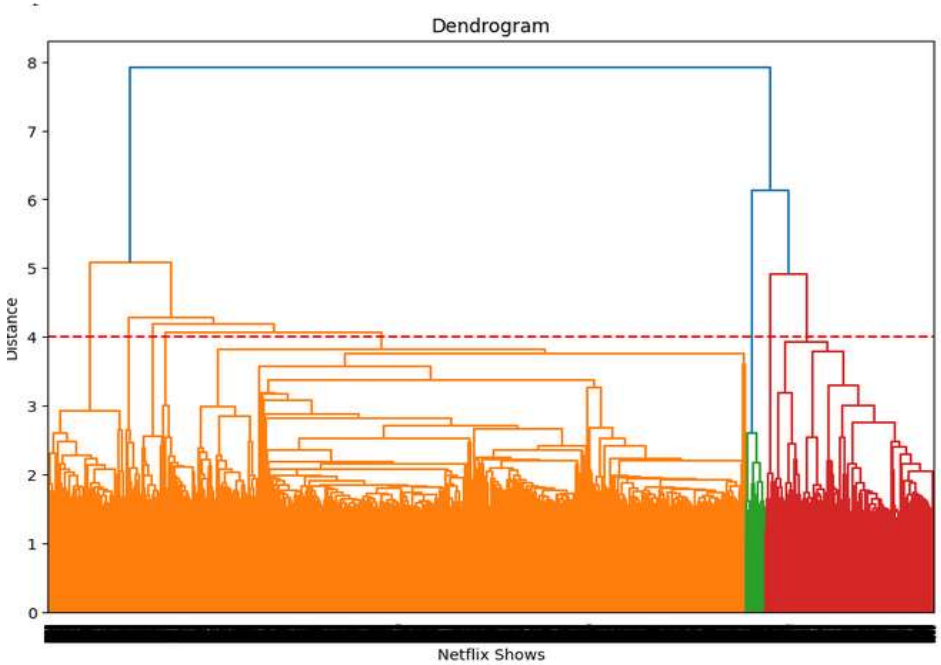
- Successfully built 4 clusters using the k-means clustering algorithm.
- In cluster 0, 1 & 3 highest number of count belong from Movie class.
- Cluster 2 mostly built on build on TV shows except 1 Movie.

Word cloud for country

Word cloud for cast

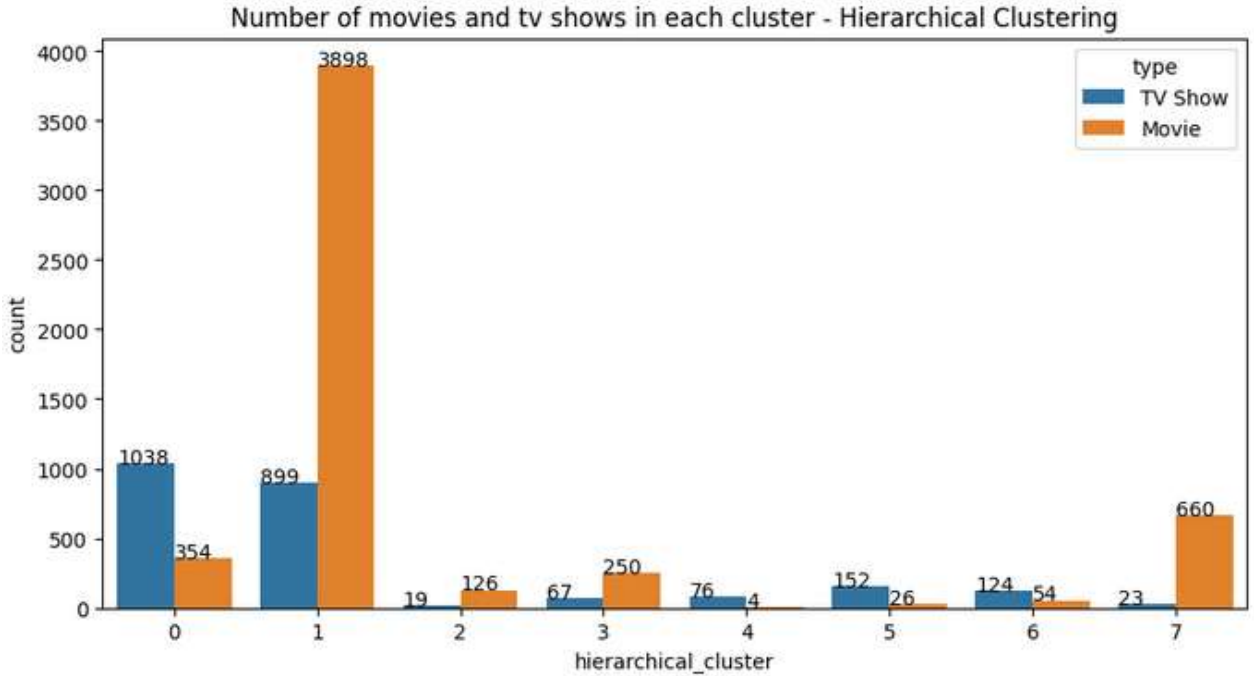
Word cloud for genre

Agglomerative Hierarchical Clustering:



- Visualizing the dendrogram to decide on the optimal number of clusters for the agglomerative (hierarchical) clustering algorithm.
- At a distance of 4 units, 8 clusters can be built using the agglomerative clustering algorithm.

Agglomerative Hierarchical clusters:



- Successfully built 8 clusters using the Agglomerative (hierarchical) clustering algorithm.
- Highest number of datapoint build on cluster 0 & 1.

Word cloud for genre

Word cloud for country

Recommendation System:

- We can build a simple content-based recommender system based on the similarity of the movie/shows.
- If a person has watched a show on Netflix, the recommender system must be able to recommend a list of similar shows that s/he likes.
- To get the similarity score of the shows, we can use cosine similarity.
- The similarity between two vectors (A and B) is calculated by taking the dot product of the two vectors and dividing it by the magnitude value. We can simply say that the CS score of two vectors increases as the angle between them decreases.

```
recommend('Naruto')
```

```
If you liked 'Naruto', you may also enjoy:

Naruto Shippûden the Movie: Bonds
Naruto Shippuden: The Movie
Naruto Shippuden : Blood Prison
Naruto the Movie 2: Legend of the Stone of Gelel
Naruto Shippûden the Movie: The Will of Fire
Naruto the Movie 3: Guardians of the Crescent Moon Kingdom
Naruto Shippuden: The Movie: The Lost Tower
Dino Girl Gauko
DRIFTING DRAGONS
Marvel Anime: Wolverine
```

```
recommend('Our Planet')
```

```
If you liked 'Our Planet', you may also en

Nature's Great Events: Diaries
Nature's Great Events (2009)
Nature's Weirdest Events
Blue Planet II
Planet Earth II
Life Story
The Making of Frozen Planet
Night on Earth
Moving Art
The Hunt
```

```
recommend('Phir Hera Pheri')
```

```
If you liked 'Phir Hera Pheri', you may also enjoy:

Bhool Bhulaiyaa
Thank You
Golmaal: Fun Unlimited
Chup Chup Ke
Bhagam Bhag
Ready
Khushi
Life in a ... Metro
Hattrick
Hasee Toh Phasee
```

CONCLUSION:

In this project, we worked on a text clustering problem wherein we had to classify/group the Netflix shows into certain clusters such that the shows within a cluster are similar to each other and the shows in different clusters are dissimilar to each other.

- The dataset contained about 7787 records, and 11 attributes. We began by dealing with the dataset's missing values and doing exploratory data analysis (EDA).

- It was decided to cluster the data based on the attributes: director, cast, country, genre, rating and description. The values in these attributes were tokenized, preprocessed, and then vectorized using TFIDF vectorizer.

- Through TFIDF Vectorization, we created a total of 10000 attributes.

- We used Principal Component Analysis (PCA) to handle the curse of dimensionality. 3000 components were
  able to capture more than 80% of variance, and hence, the number of components were restricted to 3000.

- We first built clusters using the K-Means Clustering algorithm, and the optimal number of clusters came out to be 4. This was obtained through the elbow method and Silhouette score analysis.

- Then clusters were built using the Agglomerative clustering algorithm, and the optimal number of clusters came out to be 8. This was obtained after visualizing the dendrogram.

- A content-based recommender system was built using the similarity matrix obtained after using cosine similarity. This recommender system will make 10 recommendations to the user based on the type of show they watched.

Thank You