Capstone Project - 4

Netflix Movies and TV Shows Clustering

Submitted by

Yash Dhandar

Agenda

- Problem Statement
- Data Summary
- Data Cleaning
- Exploratory Data Analysis (EDA)
- Feature Engineering
- Dimensionality Reduction
- Clustering
- Word cloud on Clusters
- Content Based Recommender System
- Challenges Faced
- Conclusions

Abstract

- Netflix is a popular streaming service and production firm.
- According to Statista, Netflix had approximately **223.09** million paid subscribers worldwide as of the third quarter of 2022.
- It is crucial that they effectively cluster the shows that are hosted on their platform in order to enhance the user experience for its subscribers.

Problem Statement

- The goal of this project is to cluster the shows on Netflix such that the shows within a cluster are similar to each other and the shows in different clusters are dissimilar to each other.
- These clusters may be later leveraged to offer the consumers **personalized show recommendations** based on their interests.
- The dataset contains **7787** records, and **11**attributes

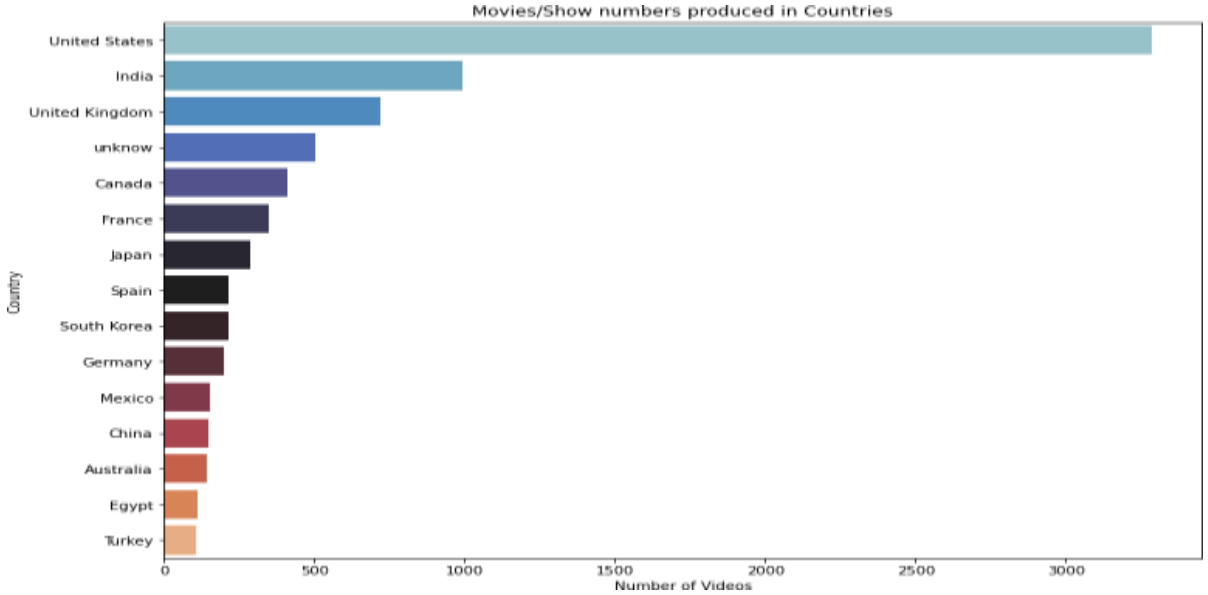
Data Summary

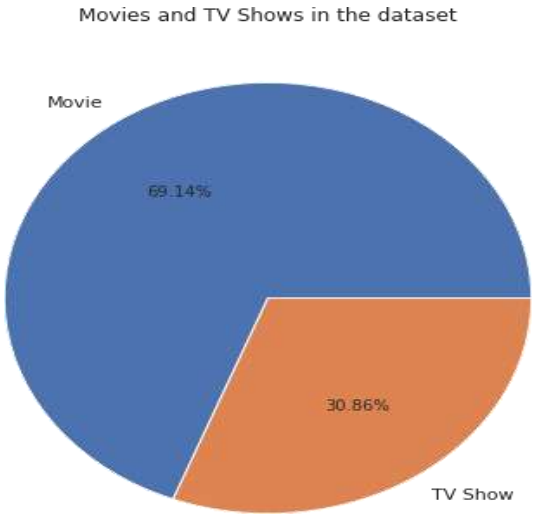
- **Show ID**
- **Type** – Movie / TV show
- **Title** – Show title
- **Director** – Name of the director
- **Cast** – Name of the cast
- **Country** – Production country
- **Date added**
- **Release year**
- **Rating** – Show age rating
- **Duration** – Minutes / seasons
- **Listed in** - Genre
- **Description**

Data Cleaning

- Handling missing values:
  - Director (2389), cast (718), and country (507) – replace with **'Unknown'**
  - Date added (10) - **dropped**.
  - Rating (7) – **mode** imputation.
- Only primary genre and country were selected to simplify the EDA
- The dataset contained separate age ratings for movies and TV shows, and were replaced with values of: 'Adults', 'Teens', 'Young Adults', 'Older Kids', 'Kids'

Exploratory Data Analysis (EDA)

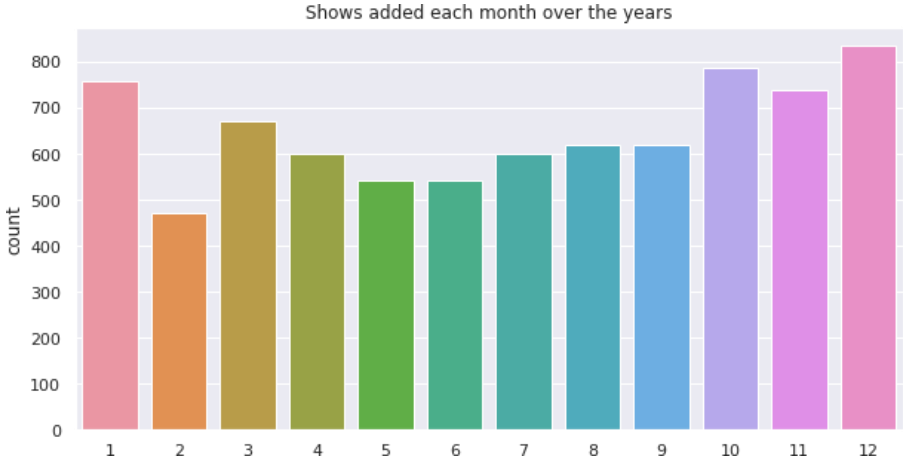


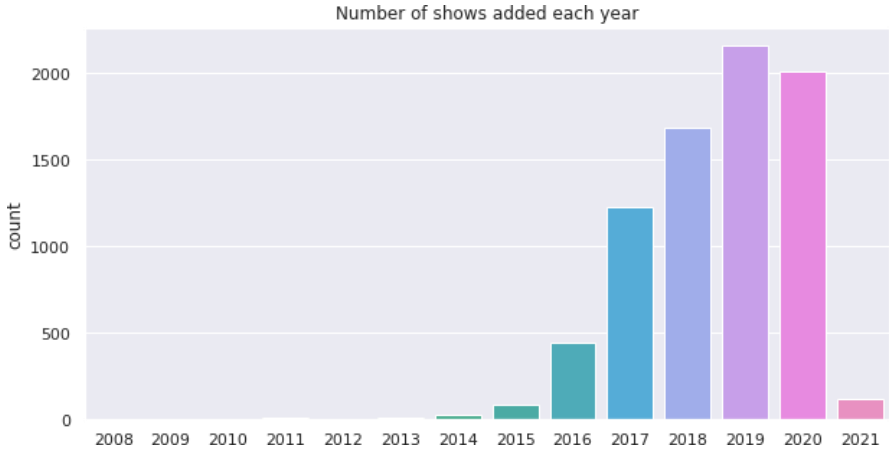


- **69.14%** of the shows on Netflix are movies, and **30.86%** TV shows.
- The top 3 countries together account for about **56 %** of all movies and TV shows in the dataset.
- This value increases to about **78%** for top ten countries.

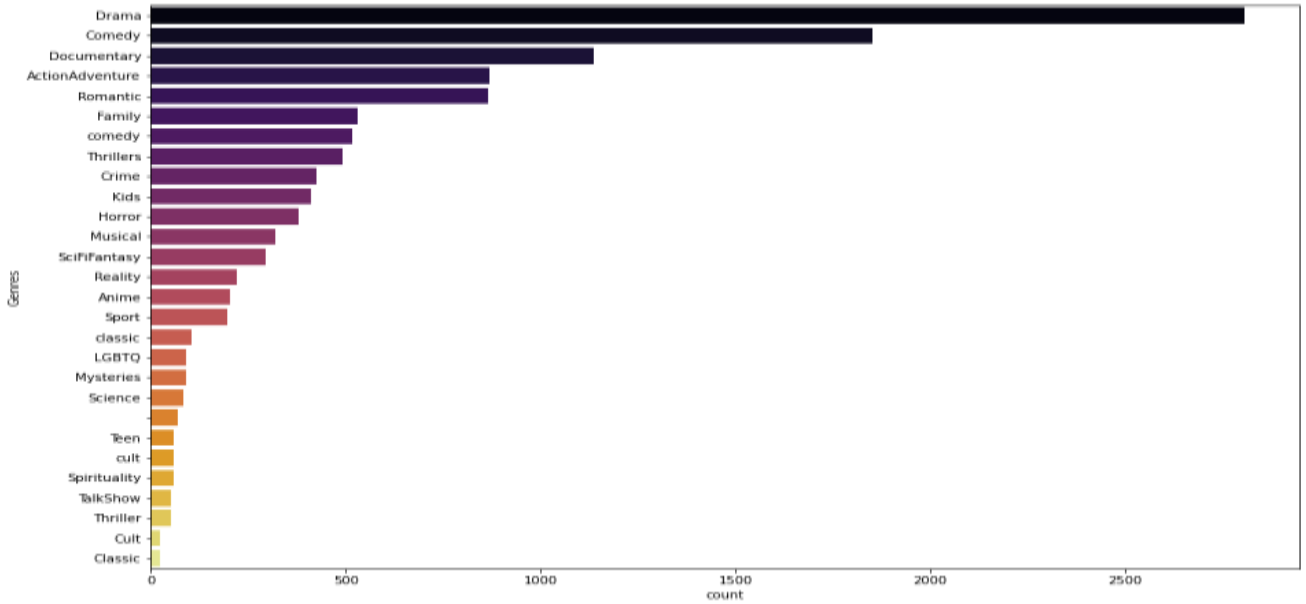
EDA (Contd.)

- More shows are added in the months of **October, November, December, and January**.
- There is a **decrease** in the number of shows added in the year **2020**, which might be attributed to the **Covid-induced lockdowns**, which halted the creation of shows.
- There are very few shows added in the year **2021**, since the data is available only up to 16th January.

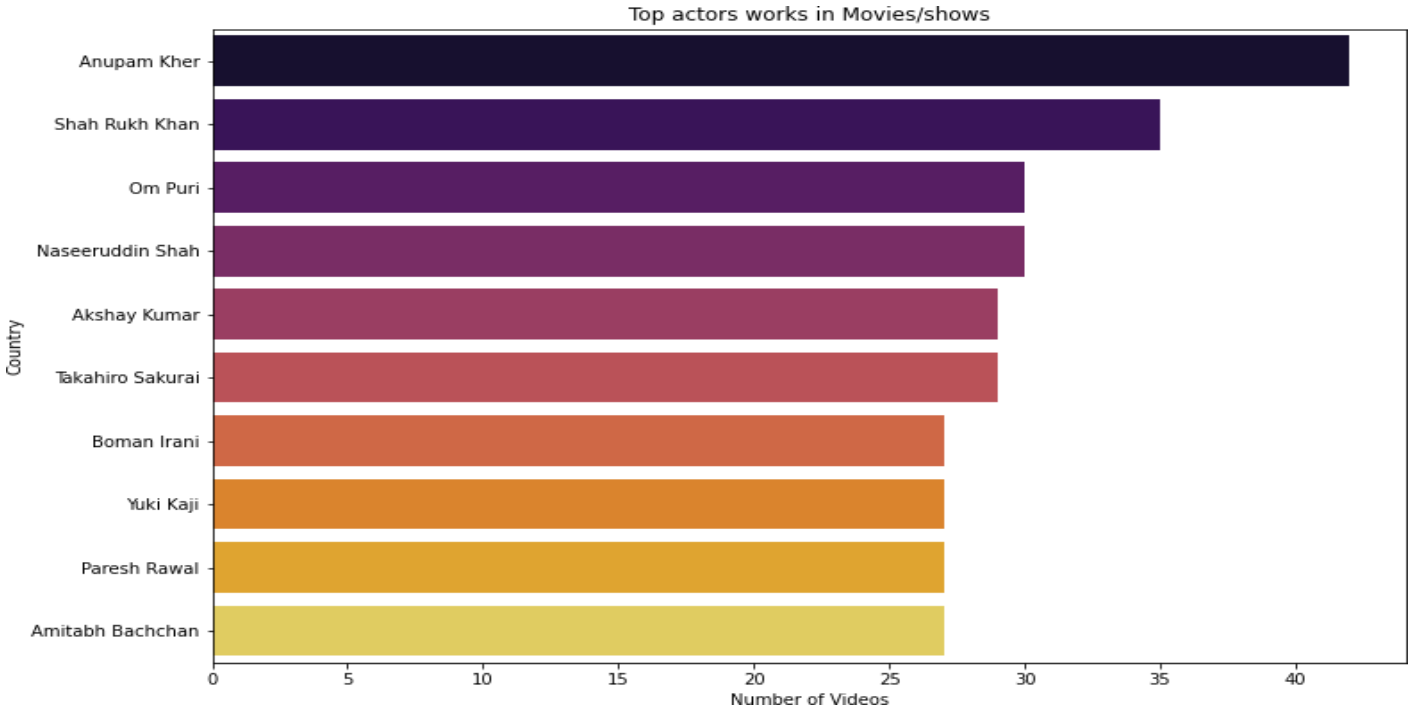




EDA (Contd.)

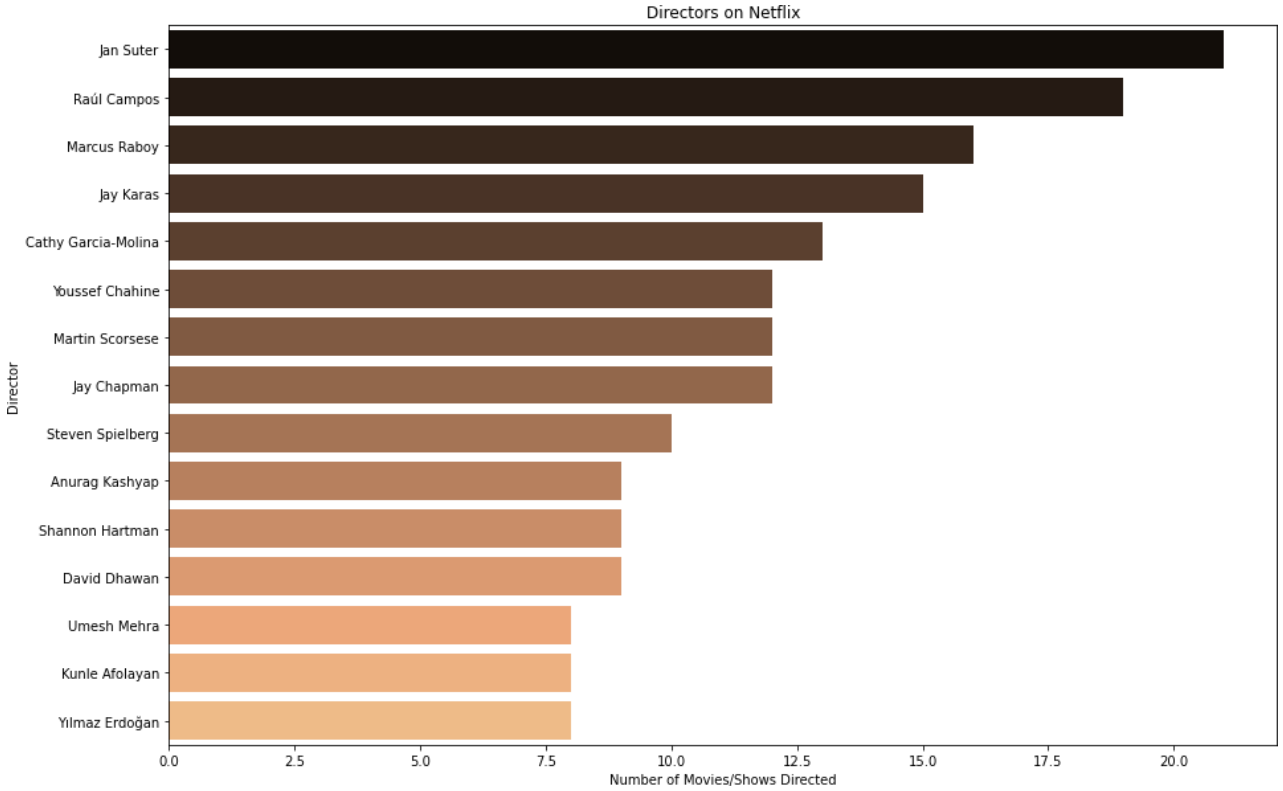


•The dramas is the most popular genre followed by comedies and documentaries. These three genres account for about 41% of all movies and TV shows. This value increases to about 82% for top 10 genres.



Top Actors on Netflix are:

1.Anupam Kher 2.Shah Rukh Khan 3.Naseeruddin Shah 4.Om Puri 5.Akshay Kumar

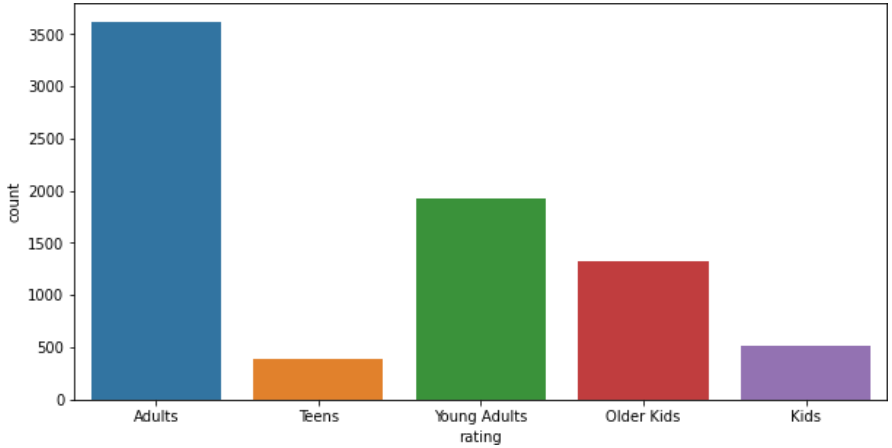


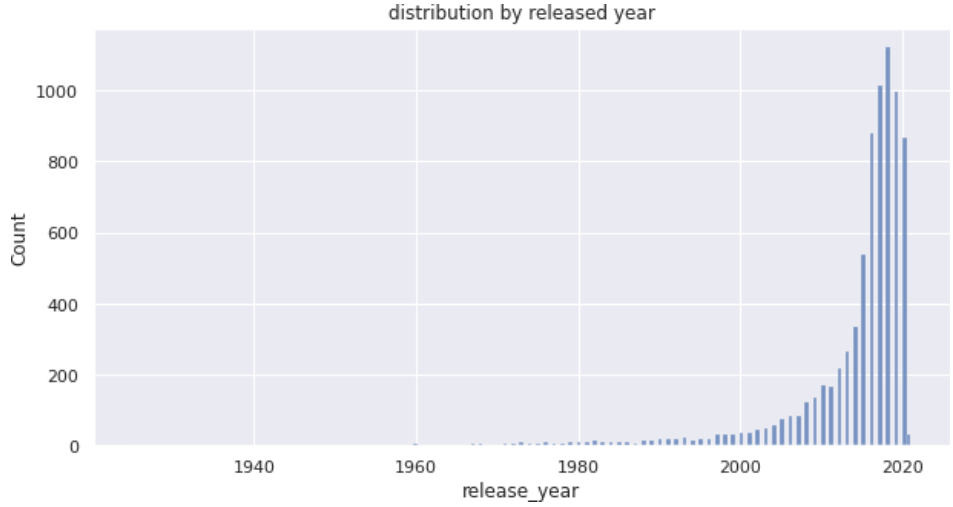
Top Directors on Netflix are:

1.Jan Suter 2.Raul Campos 3.Marcus Raboy 4.Jay Karas 5.Cathy Garcia-Molina

EDA (Contd.)

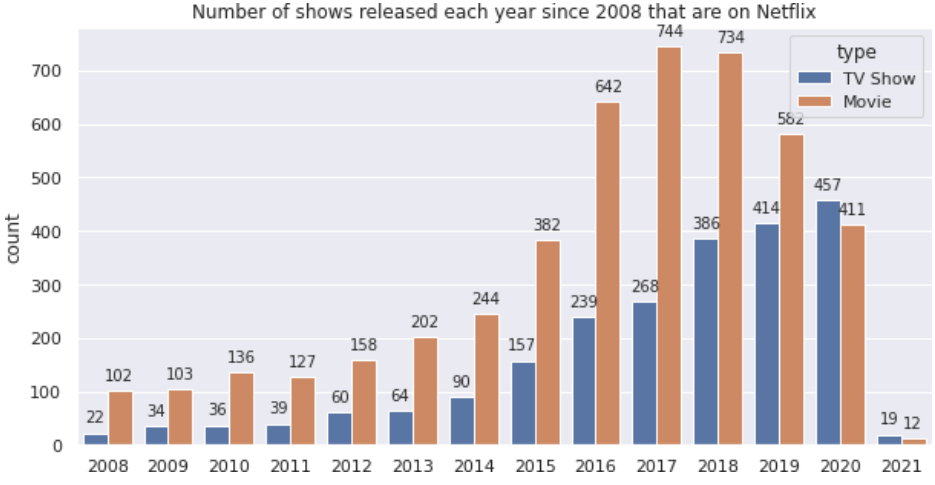
- The majority of the shows on Netflix are created to the needs of **adult** and **young adult** population.
- Netflix has greater number of **new** movies / TV shows than the old ones.

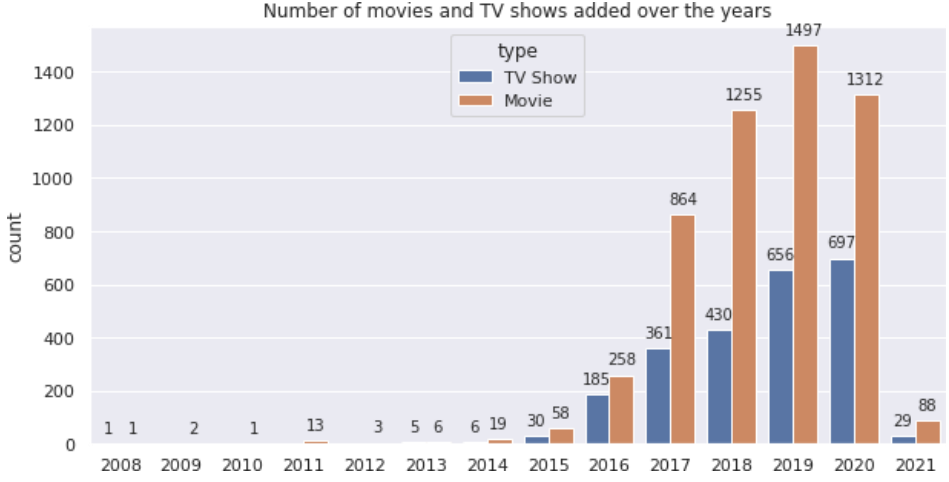




EDA (Contd.)

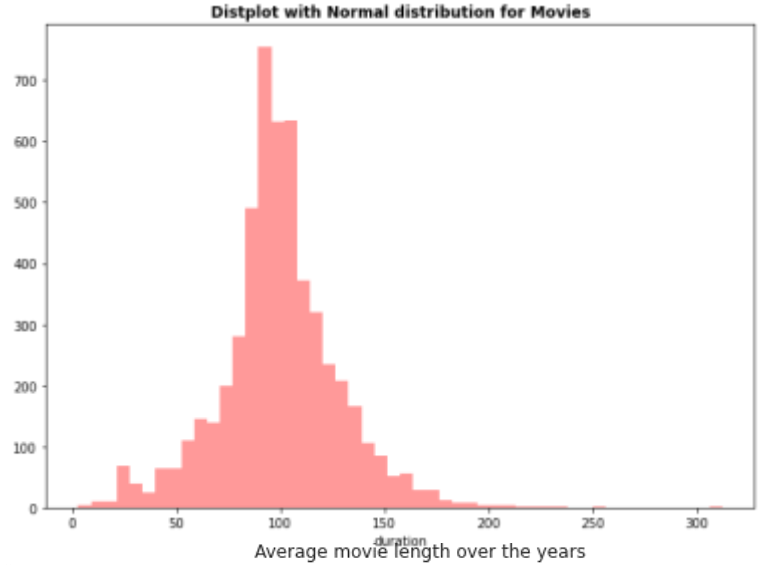
- Though there was a decrease in the number of movies added in **2020**, this pattern did not exist in the number of TV shows added in the same year.
- This might signal that **Netflix is increasingly concentrating on introducing more TV series** to its platform rather than movies.

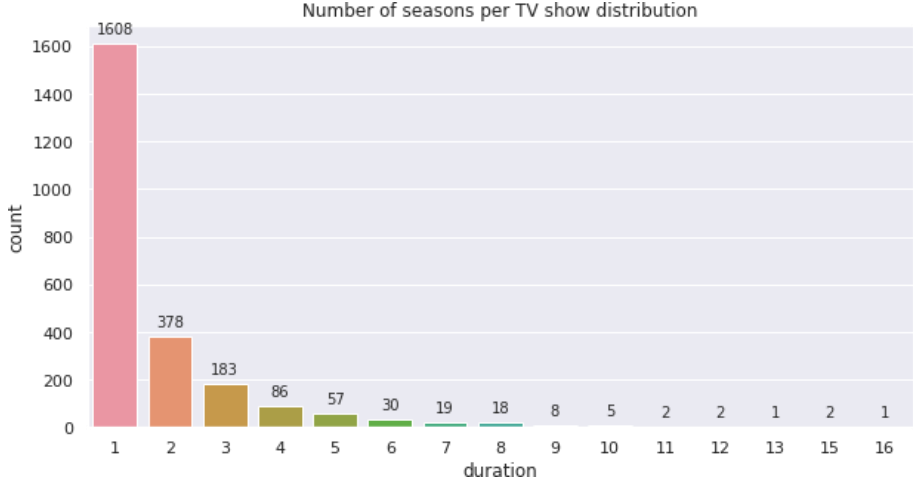


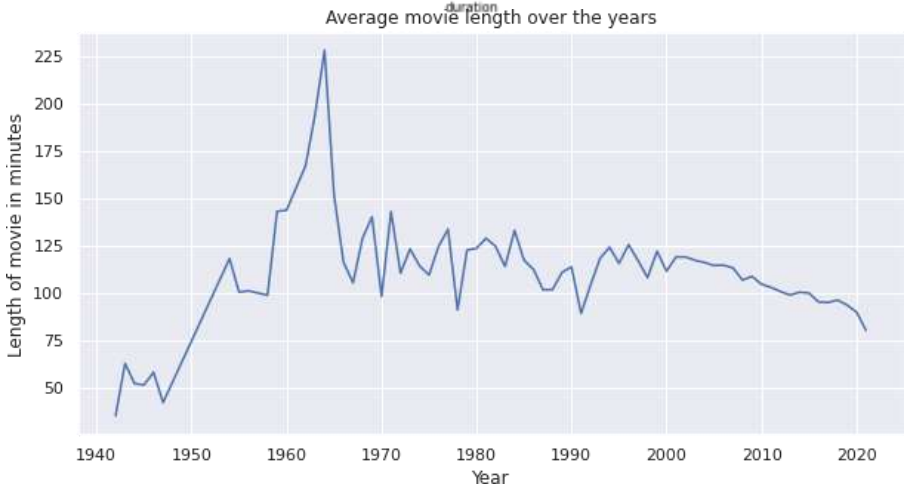


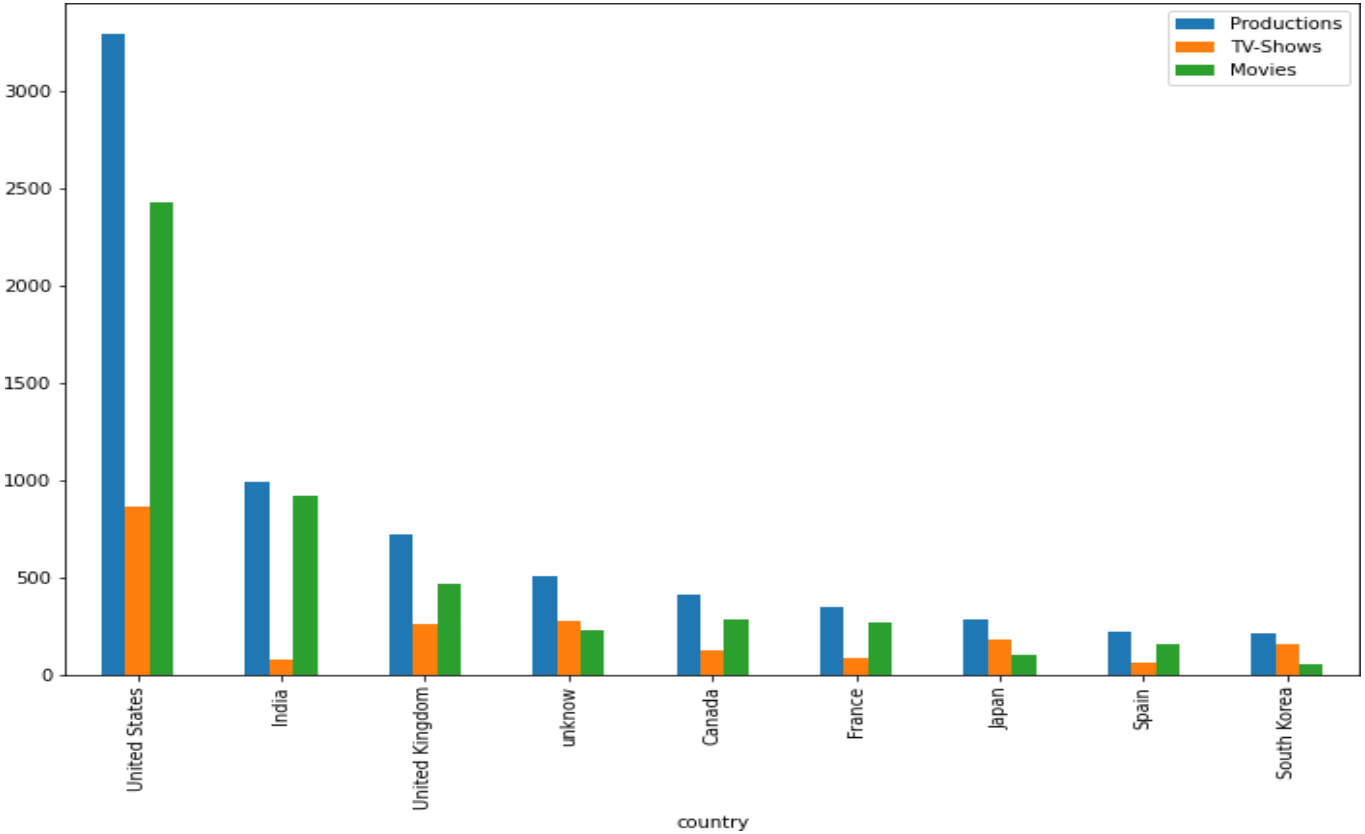
EDA (Contd.)

- The length of movies are almost **normally distributed**.
- Majority of the TV shows are still in the **1st season**.









- Production of tvshows and movies in US is higher than other countries.Production of tvshows is less in many countries but japan and south korea are seems to be interested in production of tvshows than movies.

Feature Engineering

- **Clusters are built based on the attributes:** Director, Cast, Country, Listed in (genres), and Description
- **Steps involved in data pre-processing:**
  - Removing non-ascii characters
  - Removing stop words and converting to lowercase
  - Removing punctuation marks
  - Lemmatization, tokenization and text vectorization
  - Dimensionality reduction using PCA

Feature Engineering (Contd.)

- **TFIDF** (Term Frequency Inverse Document Frequency) vectorizer was used to vectorize the corpus.

$$TF = \frac{\textbf{Number of times term t appears in a document}}{\textbf{Total number of terms in the document}}$$

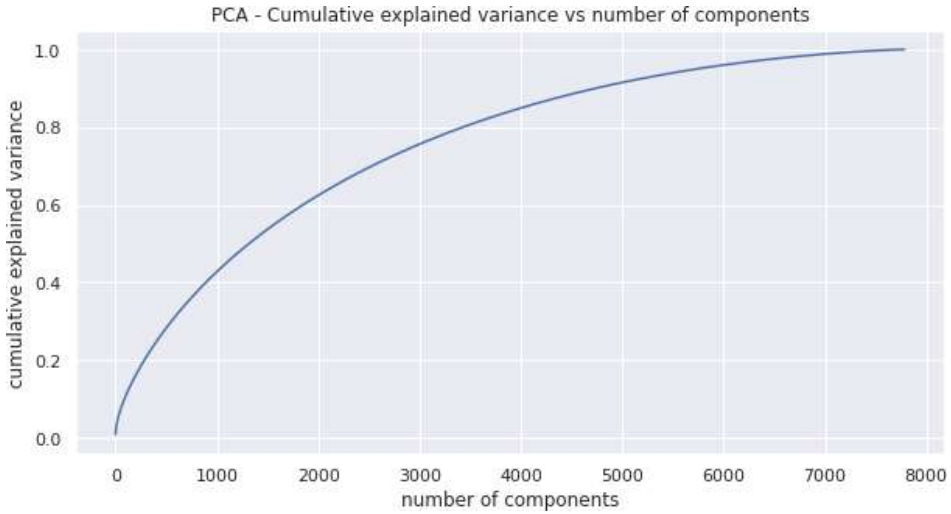
$$IDF = \log_e \left(\frac{\textbf{Total number of documents}}{\textbf{Number of documents with term t in it}} \right)$$

$$TFIDF = TF \times IDF$$

- Maximum number of features were taken as **20000**.

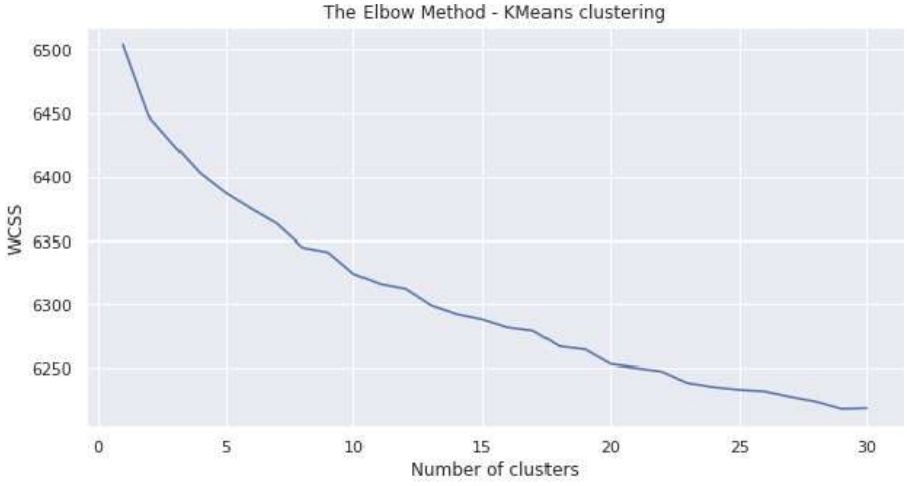
Dimensionality Reduction

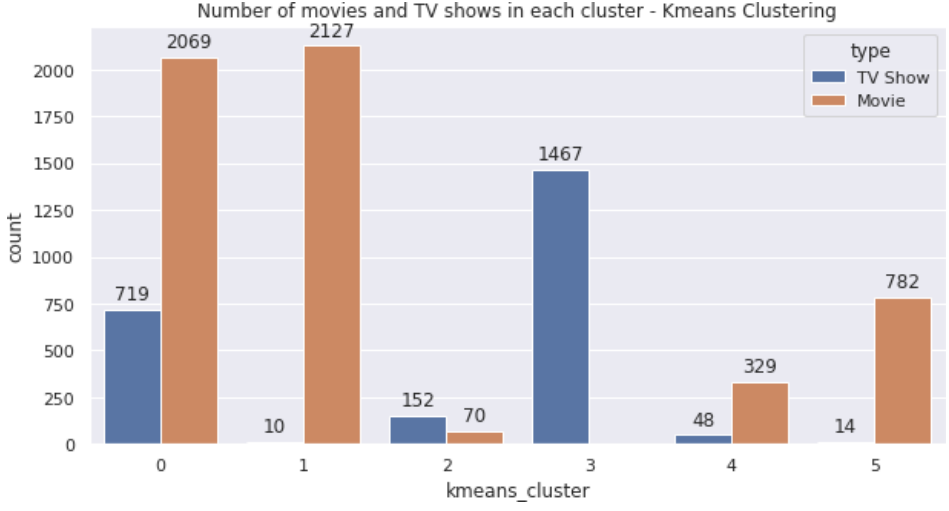
- **100%**of the variance in data is explained by about **~7500** components.
- To reduce dimensionality, only the top **4000** components were taken, which will still be able to capture more than **80%** of variance in the data.

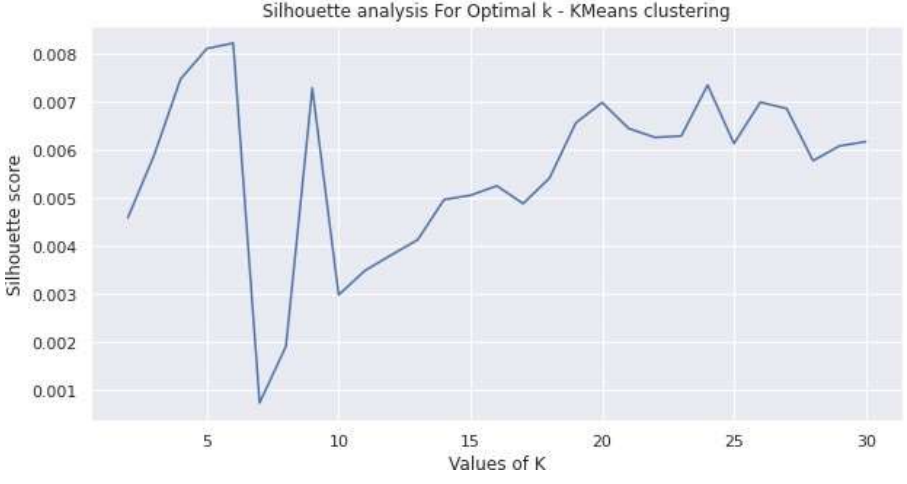


K Means Clustering

- Distortion:6374.78
- Silhouette score:0.0082
- Number of clusters:6







Word Clouds: K Means Clusters

K Means Cluster - 0

K Means Cluster - 1

K Means Cluster - 2

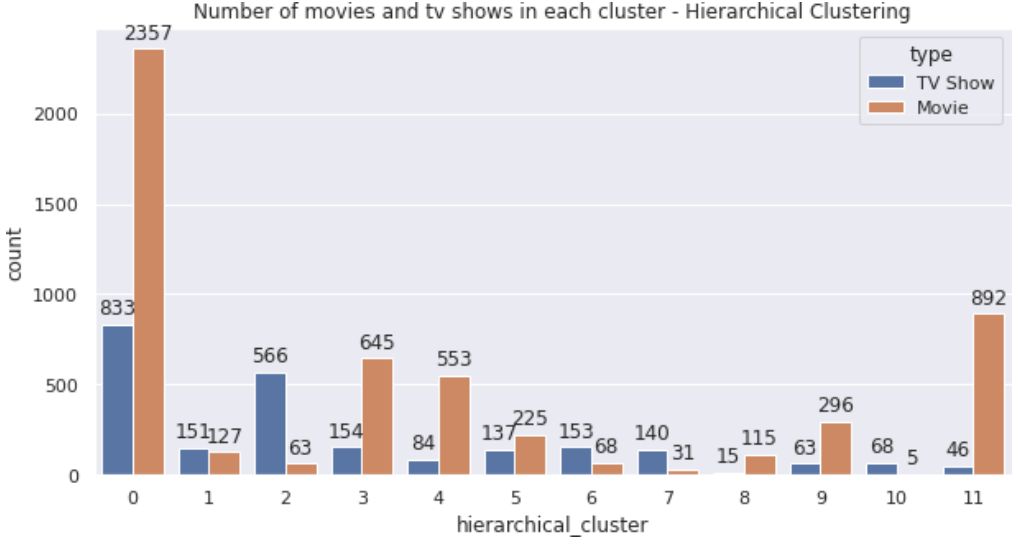
K Means Cluster - 3

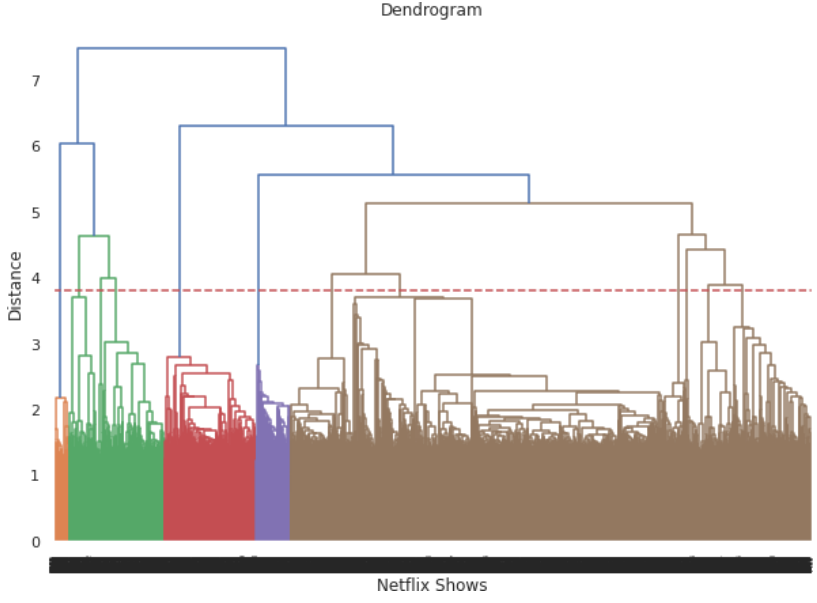
K Means Cluster -4

K Means Cluster - 5

Hierarchical Clustering

- Agglomerative clustering.
- Distance: Euclidean
- Linkage: Ward
- Number of clusters: 12





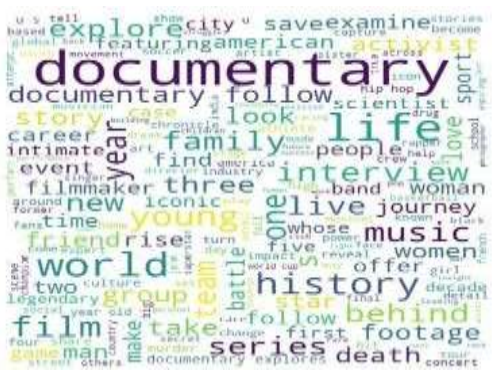
Word Clouds: Hierarchical Clusters

Hierarchical Cluster - 0

Hierarchical Cluster - 1

Hierarchical Cluster - 2

Hierarchical Cluster - 3



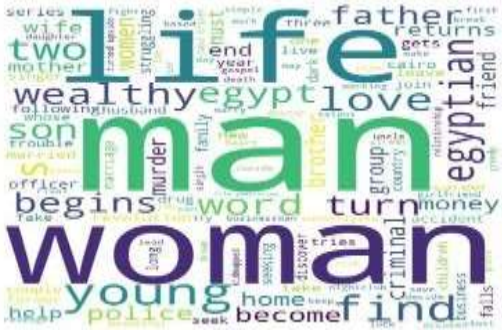
Hierarchical Cluster - 4

Hierarchical Cluster - 5

Word Clouds: Hierarchical Clusters (Contd.)

Hierarchical Cluster - 6

Hierarchical Cluster - 7



Hierarchical Cluster - 8

Hierarchical Cluster - 9

Hierarchical Cluster - 10

Hierarchical Cluster - 11

Content Based Recommender System

- We can build a simple content based recommender system based
  - on the **similarity** of the 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 Cosine Similarity score of two vectors increases as the angle
  - between them decreases.

$$\mathbf{Cos}\,\boldsymbol{\theta} = \frac{\boldsymbol{A} \cdot \boldsymbol{B}}{|\boldsymbol{A}|.\,|\boldsymbol{B}|}$$

Content Based Recommender System (Contd.)

- 10 recommendations for the show **"A Man Called God"** and **"Stranger Things"**

```
If you liked 'A Man Called God', you may also enjoy:

['Mr. Sunshine',
 'One Spring Night',
 'Rugal',
 'The King: Eternal Monarch',
 'My Mister',
 'My Little Baby',
 'Reply 1994',
 'Extracurricular',
 'My Secret Romance',
 'Chef & My Fridge']
```

```
If you liked 'Stranger Things', you may also enjoy:

['Beyond Stranger Things',
 'Prank Encounters',
 'The Umbrella Academy',
 'Haunted',
 'Scream',
 'Warrior Nun',
 'Nightflyers',
 'Zombie Dumb',
 'Kiss Me First',
 'The Vampire Diaries']
```

Content Based Recommender System (Contd.)

- 10 recommendations for the show **“Peaky Blinders”** and **“Lucifer”**

```
If you liked 'Peaky Blinders', you may also enjoy:

['Kiss Me First',
 'Happy Valley',
 'London Spy',
 'The Frankenstein Chronicles',
 'Paranoid',
 'Get Even',
 'Giri / Haji',
 'My Hotter Half',
 'The Murder Detectives',
 'I AM A KILLER: RELEASED']
```

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

['Rica, Famosa, Latina',
 'Get Shorty',
 'The Good Cop',
 'Jack Taylor',
 'Better Call Saul',
 'Dramaworld',
 'Father Brown',
 "Marvel's Iron Fist",
 'Young Wallander',
 'No Good Nick']
```

Challenges Faced

- Deciding the attributes on which we can build the clusters
- Feature engineering – deciding on the features to be dropped/kept/transformed
- Choosing the best visualization to show the trends clearly in the EDA phase
- Deciding on ways to handle the missing values
- Deciding on the attributes to be considered for clustering the dataset
- High computation time

Conclusions

- In this project, we worked on a **text clustering problem** wherein we had to cluster the Netflix shows 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 found that Netflix hosts **more movies** than TV shows on its platform, and the total **number of shows added on Netflix is growing exponentially**. Also, majority of the shows were produced in the **United States**, and the majority of the shows on Netflix were created for **adults** and **young adults** age group.

Conclusions (Contd.)

- It was decided to cluster the data based on the attributes: **director**, **cast**, **country**, **genre**, and **description**. The values in these attributes were **pre-processed**, **tokenized** and then **vectorized** using **TFIDF vectorizer**.
- Through TFIDF Vectorization, we created a total of **20000** attributes.
- We used **Principal Component Analysis (PCA)** to handle the curse of dimensionality. **4000** components were able to capture more than **80%** of variance, and hence, the number of components were restricted to **4000**.
- We first built clusters using the **k-means clustering algorithm**, and the optimal number of clusters came out to be **6**. This was obtained through the **elbow method** and **Silhouette score analysis**.

Conclusions (Contd.)

- **Hierarchical clustering** model was built using the **Agglomerative clustering algorithm**, and the optimal number of clusters came out to be **12**.This was obtained after visualizing the **dendrogram**.
- A **content-based recommender system** was built using the **Cosine Similarity score**. This recommender system will make **10** recommendations to the user based on the type of show they watch.

Thank You!