NETFLIX MOVIES AND TV SHOWS

K. YOGESH REDDY,

D. PRUTHVI RAJ

DATA SCIENCE TRAINEES,

ALMABETTER, BANGLORE.

ABSTRACT:

The dataset consists of tv shows and movies available on Netflix as of 2019. The dataset is collected from Flixable which is 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 services number of movies has decreased by more than 2000 titles since 2010, while its number of tv shows has nearly tripled. It will be interesting to explore what all other insights can be obtained from the same dataset.

Here we will perform different EDA techniques and clustering on the data to get ideal clusters for this problem. And also, we will build recommender system.

PROBLEM STATEMENT:

The data is related to Netflix movies and tv shows till 2019. It is understandable from the data that tv shows number has been tripled from 2010 and the number of movie titles has been reduced by 2000. We have to find several insights from the data using EDA and also, have to cluster the data based on the data.

INTRODUCTION:

In this project we have to perform the following

1. Exploratory Data Analysis
2. Understanding what type of content is available in different countries
3. Is Netflix focusing more on tv shows rather on movies
4. Cluster similar content by matching text data.

DATA SUMMARY:

The Provided data has 7787 observations.

The dataset contains 7787 rows and 12 columns.

1. show_id: Unique Id for every Movie/Tv show

2. type: Identifier – A movie or tv show
3. title : title of the movie/tv show
4. director : Director of the movie
5. cast : actor involved in the movie / tv show
6. country : Country where the movie/ show was produced
7. date-added: date it was added to Netflix
8. release_year: actual release year of the movie/show
9. rating: rating of the movie/ tv show
10. duration : total duration in minutes or number of seasons
11. listed_in : genre
12. description : the summary description

STEPS INVOLVED:

Exploratory Data Analysis:

We have performed several EDA techniques to clearly check how each and every feature is behaving with respect to the dependent feature y. We have also checked for null values, duplicate values luckily there were none. We have visualised several plots which gives several insights of all the features and we can see which features are more important.

Handling Null Values:

There were more null data in director and cast columns and also in country column but comparatively less than director and cast. In director column we have 2389 null values and cast has 718 null values. There were other two columns date_added and rating where 10 and 7 null data are present. For rating we have manually got the ratings and added and date-added we have removed as because it was difficult to get the information. For country column we have used mode to fill the null values.

Data Wrangling:

We have added two new features based on date_added, which are year_added and month_added. We have fixed ratings based on age group, there were 14 different types of ratings and we have transformed it to 4 different groups which are adults, teen, 7&above and kids. There were two or more country names in each row of country column so we have split the names and created new feature for the principal_country. We have added new feature genre, were the listed_in feature was split and the genre is added to genre feature.

Data Visualizations:

We have performed several visualizations on the dataset, which includes plotting null values, pie graph showing number of movies and tv shows, bar graph on ratings, pie plot on ratings and count of movies and tv shows, pie graph of principal country where it shows number of content produced by each country, histogram on principal country where it shows how many movies and tv shows produced by each country, bar plot on how much content is produced by each country on the basis of each year, distribution plots, point plot on count of each rating of

both movies and tv shows, bar graph on duration of shows, bar graph on count of top genre of movies and tv shows, bar graph on top content producing countries, word cloud on movie and tv shows genre and bar graph on tv shows duration and movie duration.

Performing NLTK operations:

We have added new_feature column where the new_feature has taken listed_in, description, type, principal_country and rating. The new_feature column is applied to different functions, where stop words are removed and converted to lowercase. The above functions are applied to listed_in, description and country features individually. We have performed stemming on new_feauture, description, listed_in and country.

Standard Scalar:

Machine learning algorithms perform better when numerical input variables are scaled to a standard range. Standard scaler is used to resize the distribution of values so that the mean of the observed values is 0 and the standard deviation is 1. It is an important technique that is mainly performed as a pre-processing step before many machine learning models, in order to standardize the range of functionality of the input dataset. Here we have performed standard scaler on listedin_len, description_len and country_len.

Training and Fitting Model:

We have used kmeans for clustering the data.

Before performing kmeans clustering we have to find the optimal cluster size for the problem, to get that we have to use silhouette score and elbow method, these two plots gives us the optimal cluster size using that we can perform kmeans clustering.

Silhouette Score:

It is a metric used to calculate the goodness of a clustering technique. Its value ranges from -1 to 1.

1: means clusters are well apart from each other and clearly distinguished.

0: means clusters are indifferent or the distance between the clusters is not significant.

-1: means clusters are assigned in the wrong way.

Silhouette Score =(b-a)/max(a,b)

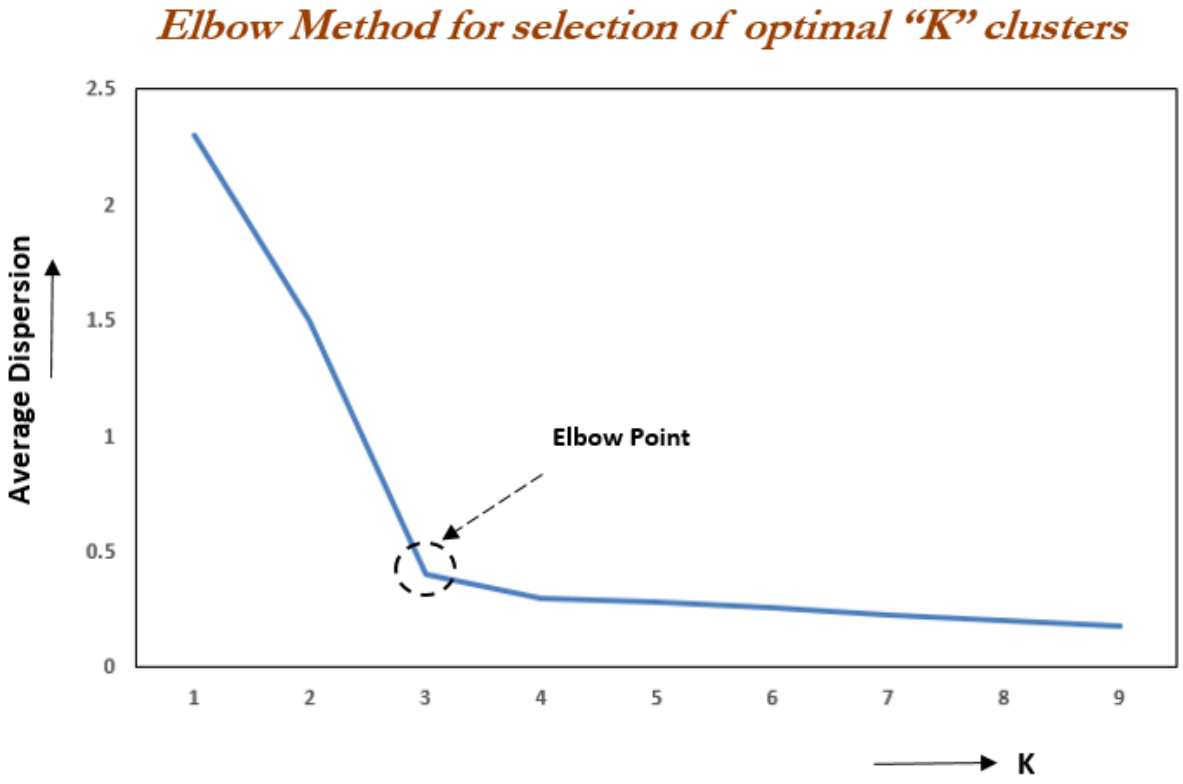
Where

a= average intra-cluster distance i.e the average distance between each point within a cluster.

b= average inter-cluster distance i.e the average distance between all the clusters.

Elbow Method:

The elbow method is used to determine the optimal number of clusters in k-means clustering. The elbow method plots the value of the cost function produced by different values of k. As you know, if k increases, average distortion will decrease, each cluster will have fewer constituent instances, and the instances will be closer to their respective centroids. However, the improvements in average distortion will decline as k increases. The value of k at which improvement in distortion declines the most is called the elbow, at which we should stop dividing the data into further clusters.



K-Means Clustering Algorithm:

It is an unsupervised learning algorithm that is used to solve the clustering problems in machine learning or data science. It groups the unlabeled dataset into different clusters. Here k defines the number of pre-defined clusters that need to be created in the process, as if k=2, there will be two clusters, and for k=3 there will be three clusters, and so on.

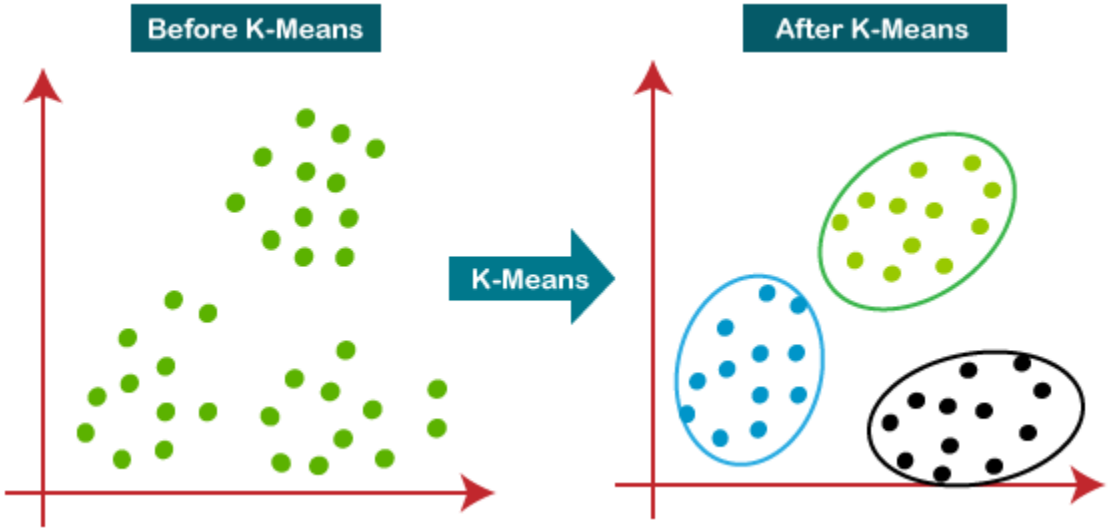
It allows us to cluster the data into different groups and a convenient way to discover the categories of groups in the unlabeled dataset on its own without the need for any training. It is centroid based algorithm, where each cluster is associated with a centroid. The main aim of this algorithm is to minimize the sum of the distances between the datapoint and their corresponding clusters.

The algorithm takes the unlabeled dataset as input divides the dataset into k-number of clusters, and repeats the process until it does not find the best clusters. The value of k should be predetermined in this algorithm.

The k-means clustering algorithm mainly performs two tasks:

- Determines the best value for k center points or centroids by an iterative process.
- Assigns each data point to its closest k-center. Those data points which are near to particular k-center, create a cluster.

Hence each cluster has datapoints with some commonalites, and it is away from other clusters.



The performance of k-means clustering depends upon highly efficient clusters that it forms. But choosing the optimal number of clusters is a big task. There are some different ways to find the optimal cluster size

- Elbow method
- Silhouette Score

We have already discussed about those two methods above.

Bag of Words:

The bag of words model is a way of representing text data when modelling text with machine learning algorithms.

The bag of words model is simple to understand and implement and has seen great success in problems such as language modelling and document classification.

A bag of words is a way of extracting features from text for use in modelling, such as with machine learning algorithms.

The approach is very simple and flexible, and can be used in myriad of ways for extracting features from documents. It is a representation of text that describes the occurrence of words within a document. It involves two things:

- A vocabulary of known words.
- A measure of the presence of known words.

CountVectorizer:

In order to use text data for predictive modelling, the text must be parsed to remove certain words- this process is called tokenization. These words need to then be encoded as integers, or floating-point values, for use as inputs in machine learning algorithms. This process is called feature extraction or vectorization.

Scikit-learn's CountVectorizer is used to convert a collection of text documents to a vector of term/token counts. It also enables the pre-processing of text data prior to generating the vector representation. This functionality makes it a highly flexible feature representation module for text.

Recommender System:

Algorithms that are used to provide the list of suggestions are called recommendation systems or engines. The recommendation system in its core algorithm uses a fundamental mathematical metric called 'similarity', which compares and quantifies the similarity between two items user selected vs rest of items in the catalog. The list of items with high similarity values to the ones that the user selected are recommended as 'you may also like'.

There are many similarity metrics used, here we have 'cosine similarity'.

Cosine Similarity ranges from -1 to 1. 1 indicates the items are same whereas -1 represents the items are dissimilar.

Conclusion:

After Implementing very good analysis and most of the visualisations we have a pretty good idea of what this dataset represents. Dataset contains 7787 Rows and 12 Columns, there are around 2800 null values in director column, 718 on cast and 507 in country.

The main content providers for Netflix are the United States which contributes 50% of the content, followed by India it contributes 14% and United Kingdom comes third of 8.5% and the rest follows.

This Netflix dataset has been categorised into two items:

- Movies
- TV Shows

Movies is the majority here which covers around 69% and TV Shows covers the rest which is 31% of the dataset.

In the Movies section 48% of the movies are Adult rated, 31% rated as Teen, 15.9% as 7&above, and the rest as Kids.

In the TV Shows section 41% of the shows are Adult rated, 27% rated as Teen, 20% as 7&above and the rest 10% as kids Rated.

The year 2017 saw the highest content ever on Netflix for movies where 744 movies were released and the yesr 2020 saw the highest for 457 TV Shows.

Our analysis on movies duration showed that movies ranging between 90mins to 150mins were the highest with 3481 movies followed by movie duration less than 90mins with 1653 movies and finally movies with more than 150mins duration has 243 movies.

Coming to the main part processing the text data using NLTK library and performing those actions on description feature. After that we have removed all the stop words and applied stemming to it (Stemming is the process of reducing a word to its word stem that affixes to suffixes and prefixes or to the roots of words known as a lemma). After that finding lengths of features and storing new features for clustering.

Standardized all the data before performing Ml Model in this case we used Kmeans algorithm because after seeing the analysis done we thought this is the algorithm can balance all the features. Plotted Silhouette for the optimal Kvalues and Elbow plot for cluster size and we found out the optimal clusters can be used are 21.

Performed Recommender systems using cosine similarity because The cosine similarity is beneficial because even if the two similar data objects are far apart by the Euclidean distance because of the size, they could still have a smaller angle between them. Smaller the angle, higher the similarity.

The recommendations after all the tuning and model selection we are getting pretty good range of recommendations and satisfied with the recommendations.

The given Netflix dataset has been successfully analysed and visualised by various plots and charts and the final result recommendations are very positive.

References:

1. Medium
2. GeeksforGeeks
3. Analytics Vidhya