# Capstone Project-4

## NETFLIX MOVIES AND TV SHOWS CLUSTERING

### Unsupervised ML  Capstone Project

Vishakha Kumari
Abhinav Akhotkar
Shivam Kumar

# INTRODUCTION

## Problem Statement-

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. It will be interesting to explore what all other insights can be obtained from the same dataset.

## Dataset-

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.

# Steps Followed

1.Exploratory Data Analysis
2.Data Cleaning & Feature Engineering
3.Data Visualization
4.Feature Engineering
5.Model Building
7.Evaluation Metrics
8.Conclusion
9. Limitations
10.Future Scope

# 1.Exploratory Data Analysis

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 : Actors involved in the movie / show
6. country : Country where the movie / show was produced
7. date_added : Date it was added on Netflix
8. release_year : Actual Releaseyear of the movie / show
9. rating : TV Rating of the movie / show
10. duration : Total Duration - in minutes or number of seasons
11. listed_in : Genere
12. description: The Summary description

## Understanding Data

- We have a dataset having 7787 records and 12 features

In this project we had following tasks:

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

## 2.Data Cleaning

- **Null values checking**
- In our dataset director, cast, country, date_added and Rating columns are having null values.
- So I filled the missing values with ‘unknown’ using fillna.

**Duplicated values**

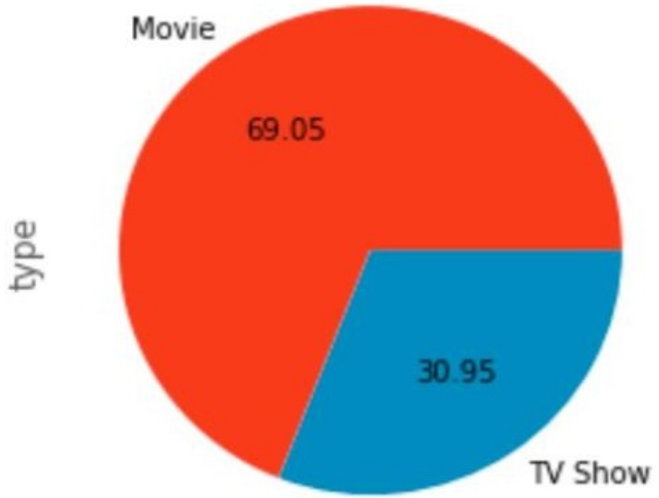
- In our dataset we don’t have any duplicate records.

## Feature Engineering for data visualization

- Changed the name of listed_in to ‘genres’ which will convenient for me in further operations.
- Then I changed data type of date added column to datetime and then I created separate columns for day, month and year by extracting dates from date added column.
- After that I dropped date added column as we extracted day, month and year.

# Data Visualization-Univariate Analysis

## Type of Content

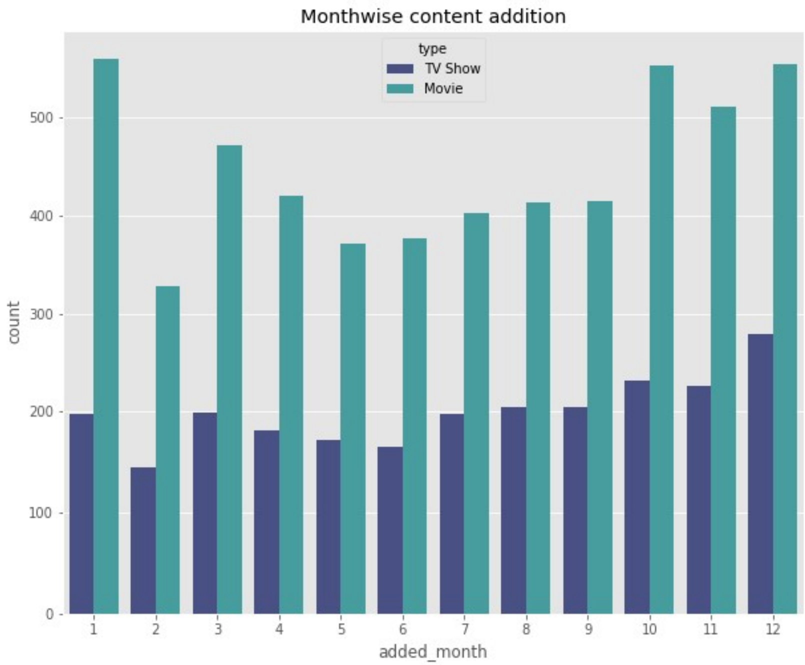


# Genres

## Tree Map

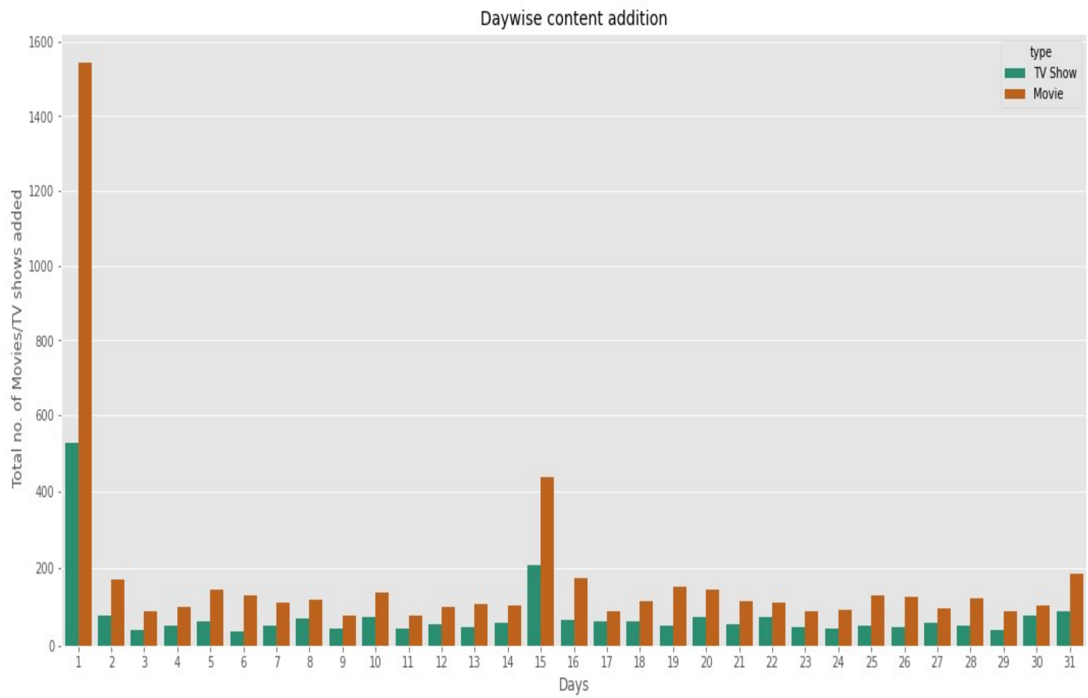
# Content Addition

## Month wise



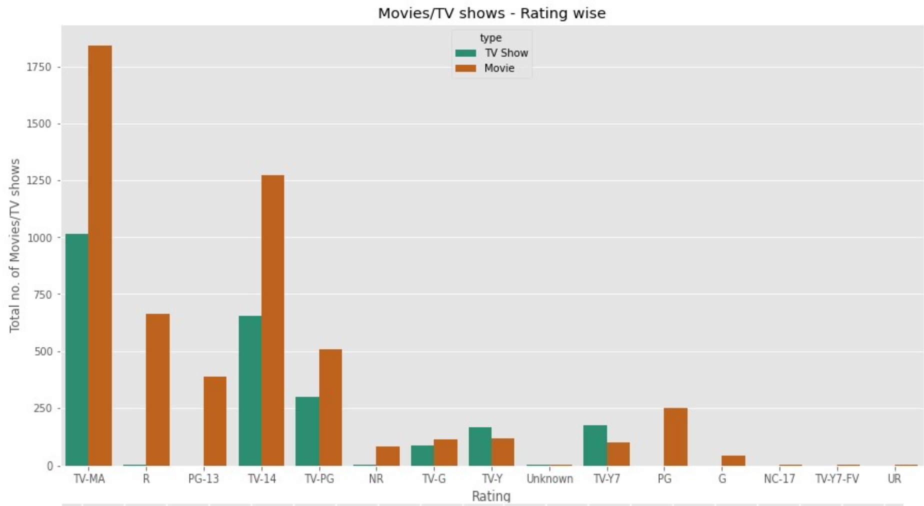

"**January**", "**October**" and "**December**" there is more content added on netflix
& "**February**" very less amount of content added.

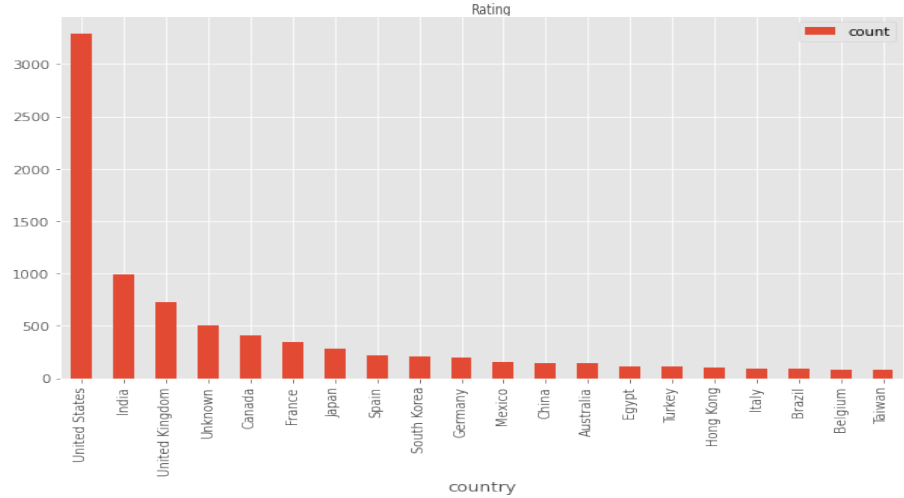
## Day wise




Netflix majorly add the content on the **1st day of every month**.
And there is another day which is **15th**, mid of the month which has second highest number of contents added on it.

# Ratings




**Netflix Rating of Movies/TV Shows based on content:-**

**TV-MA** :for Mature Audiences

**R** : Restricted

**PG-13** : Parents strongly cautioned. May be Inappropriate for ages 12 and under

**TV-14** : Parents strongly cautioned. May not be suitable for ages 14 and under

**TV-PG** : Parental Guidance suggested

**NR** : Not Rated

**TV-G** : Suitable for General Audiences

**TV-Y** : Designed to be appropriate for all children

**PG** : Parental Guidance suggested

**G** : Suitable for General Audiences

**NC-17** : the content isn't suitable for children under 17 and younger

**TV-Y7-FV** : Suitable for ages 7 and up

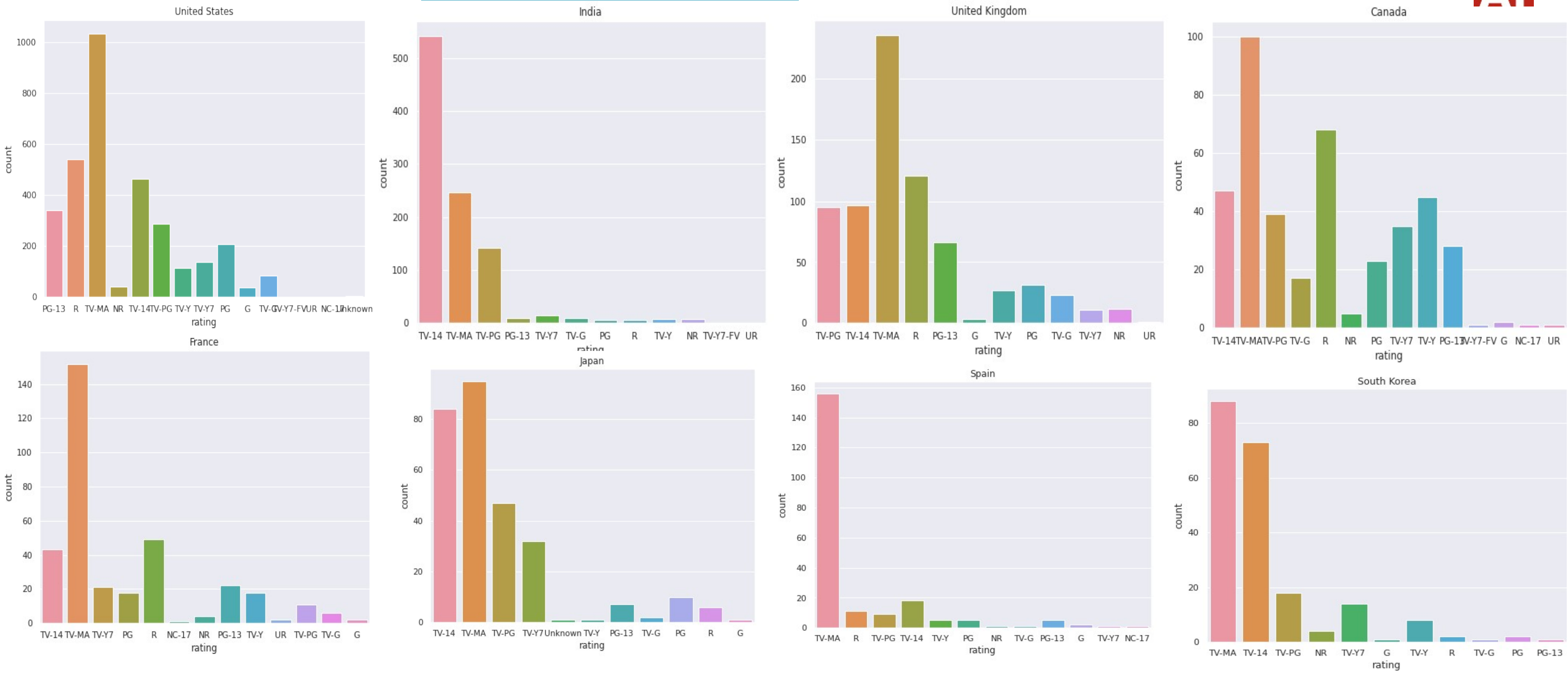
**UR** : Unrated

Most of content for **mature adults (TV-MA)** is more in both TV shows and Movies.
Then for **Under 14** which is **TV-14** is second highest in number, in which **parents are strongly cautioned.**

**G** rating which means **suitable for general audience** is rated to **very very less number of content**, which means there is **very less content which can suitable for everyone**.

Country

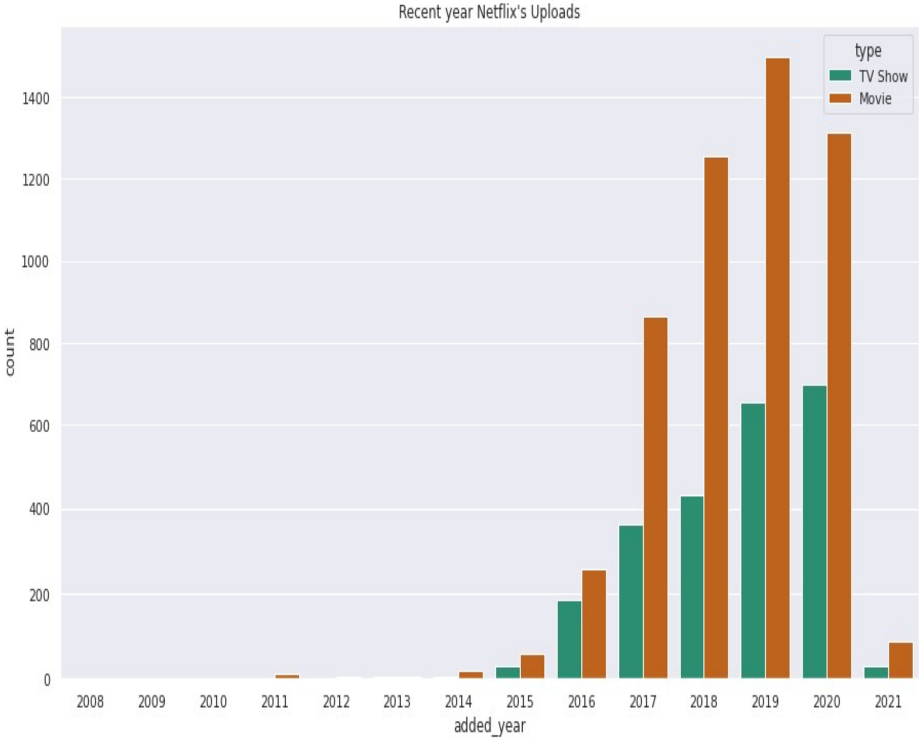
# Ratings VS country




**Here we can say that shows with TV-MA rating are highest in Belgium, Brazil, Italy, Turkey, Australia, Mexico, Germany, South Korea, Spain, Japan, France, Canada, United Kingdom, United States**

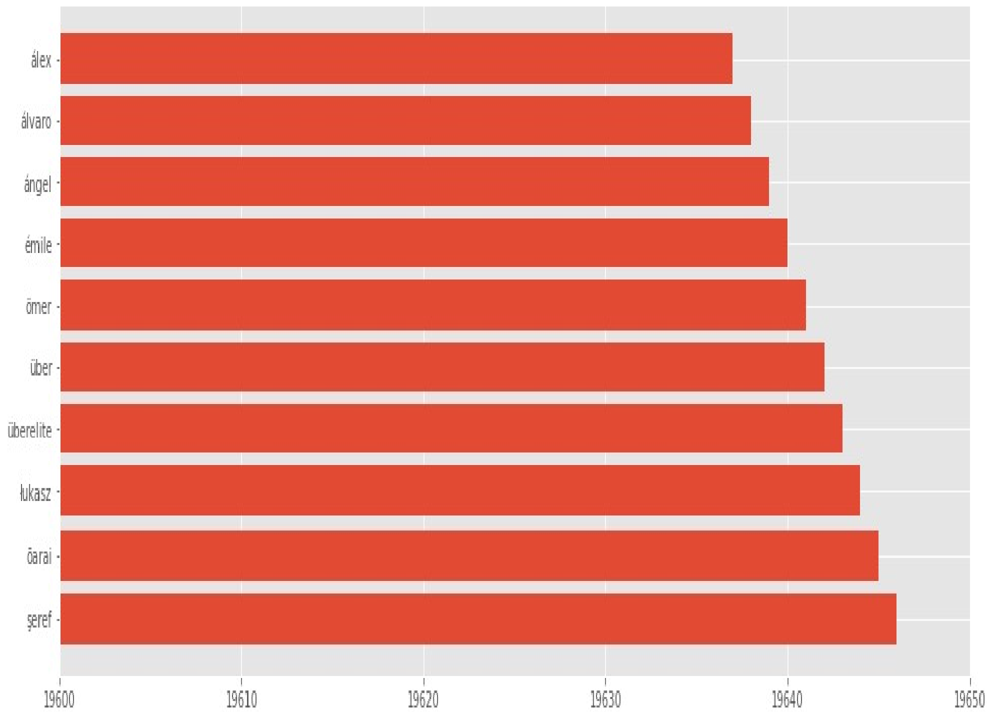
**India, China, Egypt, Hong Kong and Taiwan has highest number shows rated as TV-14**

# Netflix's focus in recent years




From above count plot we can clearly see that from 2017 number of Movies added increased tremendously, but at the same time TV shows added from 2017 are also increased but as comparison to Movies they are very less in numbers.

# Text Visualization




- In above bar chart we can see that most occurred words are non-English.

# 4.Feature Engineering

**STEMMING** - 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

**TF-IDF** - Term frequency-inverse document frequency is a text vectorizer that transforms the text into a usable vector.

**TF** - The term frequency is the number of occurrences of a specific term in a document. Term frequency indicates how important a specific term in a document. Term frequency represents every text from the data as a matrix whose rows are the number of documents and columns are the number of distinct terms throughout all documents.

**IDF** - Document frequency is the number of documents containing a specific term. Document frequency indicates how common the term is. Inverse document frequency (IDF) is the weight of a term, it aims to reduce the weight of a term if the term's occurrences are scattered throughout all the documents.

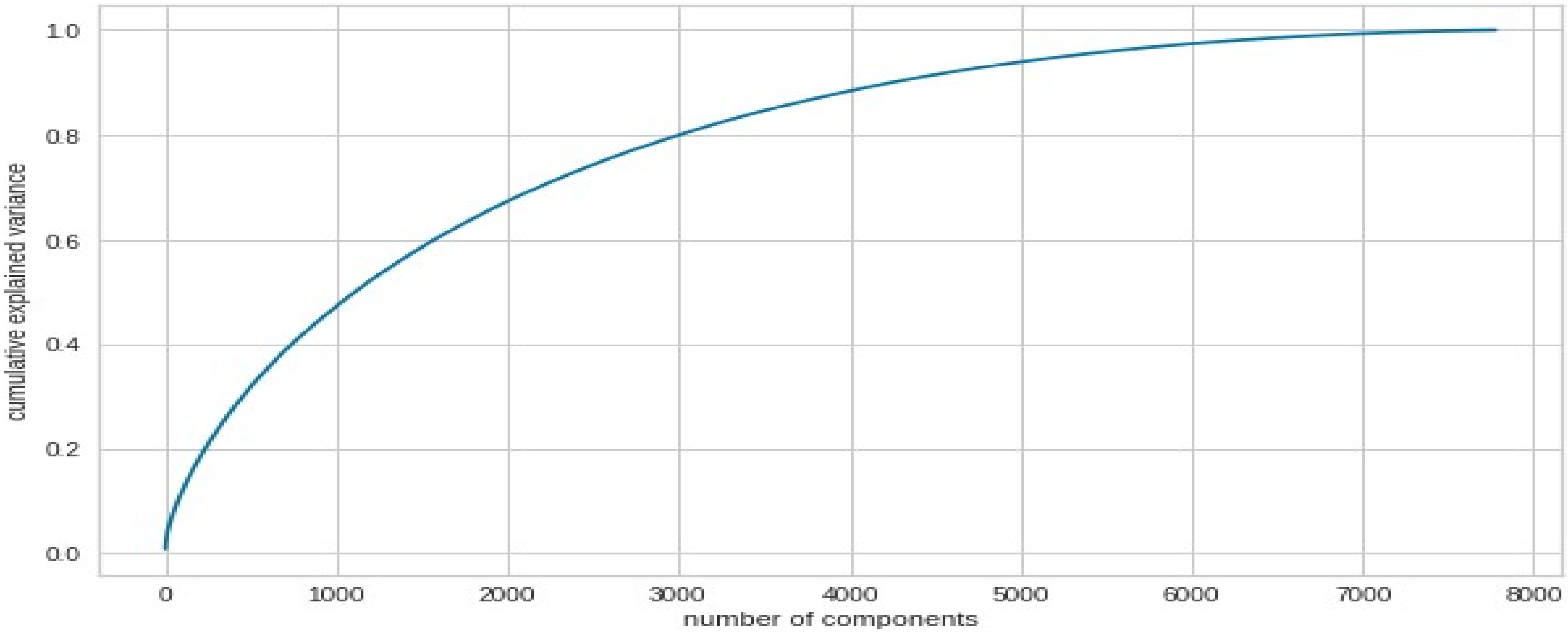
## PCA- Principal Component Analysis

Principal Component Analysis is an unsupervised learning algorithm that is used for the dimensionality reduction in machine learning. It is a statistical process that converts the observations of correlated features into a set of linearly uncorrelated features with the help of orthogonal transformation.

It is a technique to draw strong patterns from the given dataset by reducing the variances.

PCA generally tries to find the lower-dimensional surface to project the high-dimensional data.

# Cumulative Explained Variance



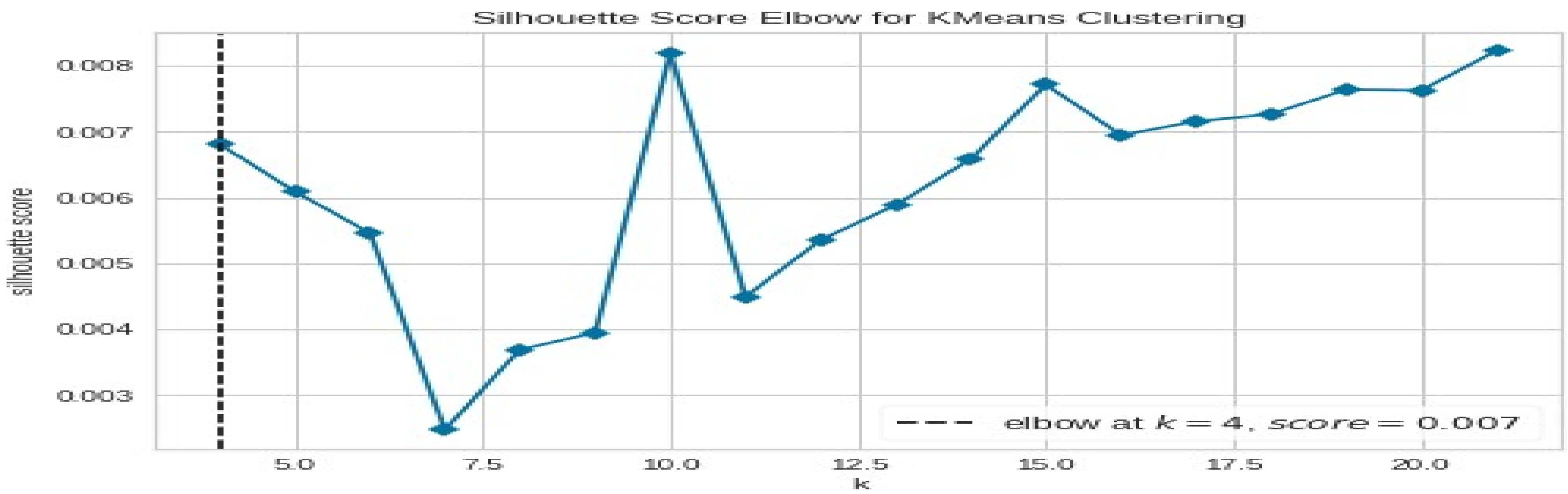

- **Here we can clearly spot that 80% variance is explained by 3000 components only.**

# 5. Model Building

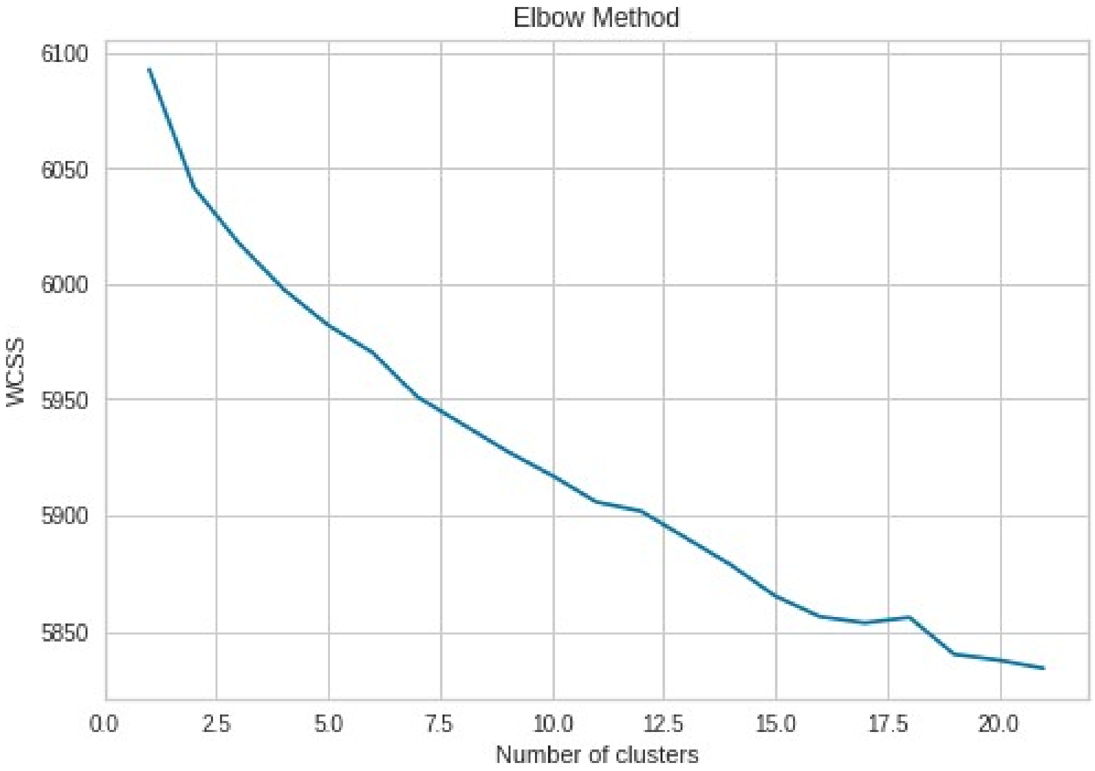
## KMeans Clustering-

K-Means Clustering is an Unsupervised Learning algorithm, which groups the unlabelled dataset into different clusters.

It is an iterative algorithm that divides the unlabelled dataset into k different clusters in such a way that each dataset belongs only one group that has similar properties.



# Elbow Method to get number of clusters




**We will take no. of clusters as k=15.**

- The K-Elbow Visualizer implements the “elbow” method of selecting the optimal number of clusters for K-means clustering.

- The elbow method runs k-means clustering on the dataset for a range of values for k (say from 1-10) and then for each value of k computes an average score for all clusters. By default, the distortion score is computed, the sum of square distances from each point to its assigned centre.

# Model Fitting-

```
# fitting the k means algorithm on lower features

kmeans= KMeans(n_clusters=15, init= 'k-means++',max_iter=300, n_init=1)
kmeans.fit(X)
```

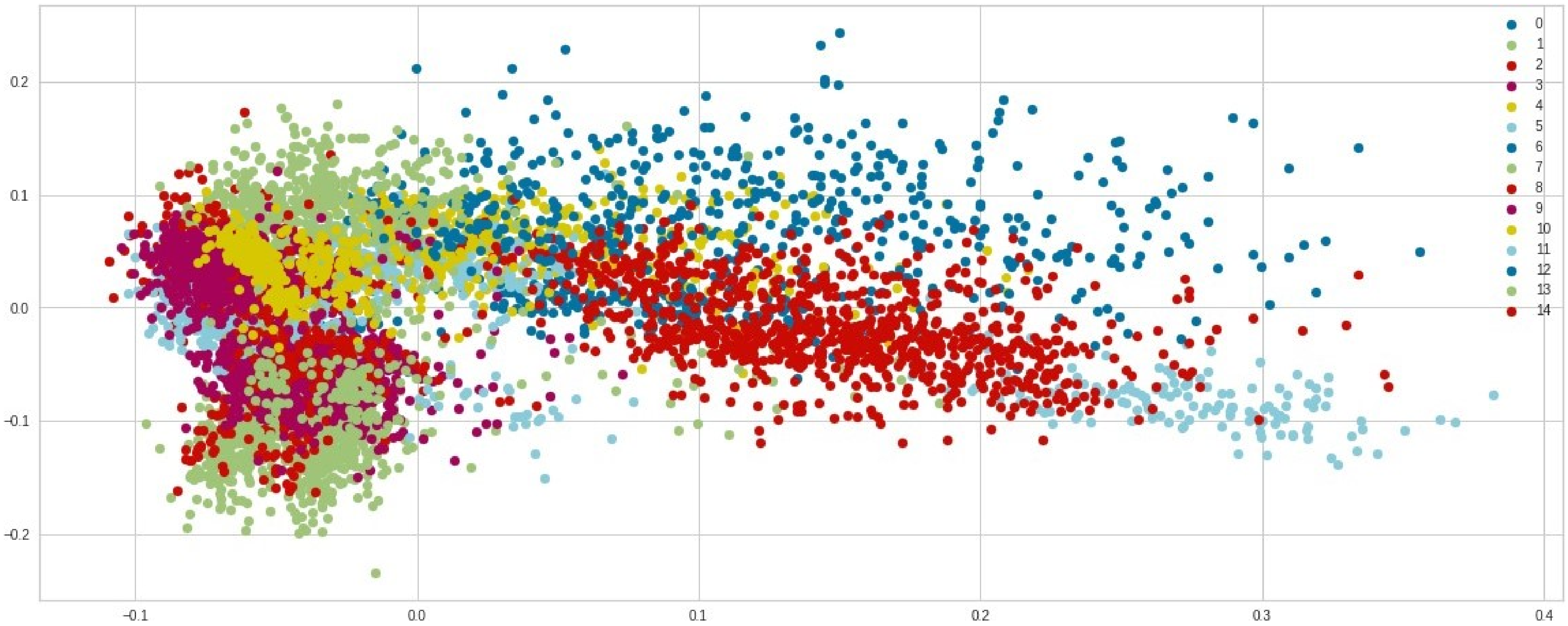
```
KMeans(n_clusters=15, n_init=1)
```

**Calculated the silhouette score for k=15 which is around 0.008**

# Predicting –

```
#predict the labels of clusters.
label = kmeans.fit_predict(X)
```

# Predicted clusters visualization-



# 6. Evaluation metrics

**Silhouette Score -**

Silhouette analysis can be used to study the separation distance between the resulting clusters. The silhouette plot displays a measure of how close each point in one cluster is to points in the neighbouring clusters and thus provides a way to assess parameters like number of clusters visually. This measure has a range of [-1, 1].

```
For n_clusters = 2 The average silhouette_score is : 0.0042239286506324325
For n_clusters = 3 The average silhouette_score is : 0.005428348884791592
For n_clusters = 4 The average silhouette_score is : 0.006645116698248495
For n_clusters = 5 The average silhouette_score is : 0.007109649705147324
For n_clusters = 6 The average silhouette_score is : 0.001533400590265247
For n_clusters = 7 The average silhouette_score is : 0.0025564559531292804
For n_clusters = 8 The average silhouette_score is : 0.006402479918956498
For n_clusters = 9 The average silhouette_score is : 0.0036254469777389103
For n_clusters = 10 The average silhouette_score is : 0.005379304723473815
For n_clusters = 11 The average silhouette_score is : 0.005105497252050331
For n_clusters = 12 The average silhouette_score is : 0.00569289785570323
For n_clusters = 13 The average silhouette_score is : 0.006647530992500081
For n_clusters = 14 The average silhouette_score is : 0.0071541422746304395
For n_clusters = 15 The average silhouette_score is : 0.0070827463187556295
```

**We selected number of clusters as 15 which in above calculations showing 0.00708 as silhouette score.**

# 7. Conclusion

1.In cumulative explained variance graph we got 80% of variance captured by 3000 components only, that's why we selected no. of components as 3000.

2. We selected no. of clusters as 15 from Elbow Method.

3. Calculated silhouette score for 15 no. of clusters which was showing 0.008

4. Then we applied KMeans on our data and then we predict the labels.

5. We plotted word cloud for each cluster so that we can visualize the summary of each cluster.

6.Then we plotted average silhouette score for clusters ranging from 2 to 16, and in that we get silhouette score 0.00708 for cluster=15 which is pretty close to earlier we calculated.

# 8. Limitations

1. As the number of dimensions increases, a distance-based similarity measure converges to a constant value between any features.

2. Centroids can be dragged by outliers, or outliers might get their own cluster instead of being ignored.

3. More Computational power required.

4.k-means has trouble clustering data where clusters are of varying sizes and density.

# 9. Future Scope

1. With more computational power can work on more data.

2. Can apply different clustering algorithms.

**Thank You !**