**Capstone Project Submission**

**On**

**NETFLIX\_MOVIES\_AND\_TV\_SHOWS\_CLUSTERING**

|  |
| --- |
| **Team Member’s Name, Email and Contribution:** |
| * 1. Ashutosh Sharma (ashutosh0626@gmail.com)      1. Data Preprocessing      2. Model Development         + Clustering         + Classification      3. Model Evaluation      4. Conclusions   2. Prashant Gour (gourprashant787@gmail.com)      1. Feature Engineering      2. Model Development         + Preprocessing |
| **Please paste the GitHub Repo link.** |
| GitHub Link: - <https://github.com/ashutosh-sharma-xi/Netflix-Movies-and-TV-Shows-Clustering> |
| **Please write a short summary of your Capstone project and its components. Describe the problem statement, your approaches, and your conclusions. (200-400 words)**   |  | | --- | | This dataset consists of tv shows and movies available on Netflix as of 2019. The dataset is collected from Flexible 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. It will be interesting to explore what other insights can be obtained from the same dataset.  Integrating this dataset with other external datasets such as IMDB ratings and rotten tomatoes can also provide many interesting findings.   * 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. * Missing values and duplicate values were handled adequately (features like Director was having around 30% cast:9.22, country: 6.51% of missing values.) * Outlier detection and removal techniques were performed to, on date added and released date features which are kind of correlated feature (61 outliers removed from date\_added feature) * We began by dealing with the dataset's missing values and doing exploratory data analysis (EDA) and found various insights including: * In our dataset we have around 69% content as movies, remaining 31% as TV shows, this signifies people generally prefer movies over TV-Shows. * Most content on Netflix is rated for Mature Audiences and is over 14 years old. * Documentaries are the topmost genre available on Netflix, followed by Stand-up and Drama * From the year 2015 there was a significant rise in the rate of making of TV -shows but in 2021 this trend discontinued due to the pandemic situation by covid-19 and further the data is not available. * Most vital & Popular words recapitulated in TV shows and movies are ‘Love', 'Christmas', 'World', 'Story', 'Man', 'Live', 'Girl', and 'Life’ these means mostly in movies and TV-Shows contains content related to these keywords. * Top countries with the greatest number of TV-Shows are  1. USA – Not highest in population but the highest content Creator 2. India 3. UK 4. Japan 5. South Korea   ---------------------------Conclusion on Modelling-----------------------------   * In this project, we worked on a text clustering problem wherein we had to classify/group the Netflix shows and movies into certain clusters such that the shows and Clusters within a cluster are similar to each other and the shows in different clusters are dissimilar to each other. * 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, the 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. * It was decided to cluster the data based on the attributes: director, cast, country, genre, and description. The values in these attributes were tokenized, pre-processed, and then vectorized using the TFIDF vectorizer. * Through TF-IDF Vectorization, we created a total of 20000 attributes. * We used Principal Component Analysis (PCA) to handle the curse of dimensionality. 3000 components were able to capture more than around 90% of the variance, and hence, the number of components was restricted to 3000. * We first built clusters using the k-means clustering algorithm, and the optimal number of clusters came out to be 10 with a good WCSS Value and high silhouette score of 0.0483. 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 20 at a distance of 20. This was obtained after visualizing the dendogram. * DBSCAN cluster didn't give satisfying results. it was a kind of biased model that clustered most of the data into a single cluster. Estimated number of clusters: 19 Estimated number of noise points: 7134 * Build an SVM Classification model with hyperparameter tuning at the end to predict a cluster number for a data record. * Classifier predicts good results with above 99% accuracy | |