Classification and Clustering Movie Reviews Data

Dhaval Patel

Northwestern University

Abstract

The goal of this project is to improve a corpus-wide vocabulary and get some insights from 200 movie reviews using Natural Language Programming (NLP) methodologies. We will evaluate several clustering, classification, and topic modeling experiments for this assignment. The following NLP techniques such as k-means clustering, Latent Semantic Analysis (LDA), Latent Dirichlet Allocation (LDA), and classification models such as Naïve Bayes, Support Vector Machine (SVM), and Random Forest are used in these experiments. The purpose is to examine how these various methods categorize documents into distinct categories such as positive/negative reviews, genres, movie titles, or previously unconsidered classifications.

Introduction

Clustering is the process of separating data points into groups so that data points in the same group are comparable to other data points in the same group but different to data points in other groups. The objective is to separate groups with similar characteristics and assign them to clusters. Clustering is a straightforward method for doing several simple analysis and gaining rapid insights into data from various domains. Cluster analysis is used by the retail industry to swiftly segment client demographics, by the insurance industry to quickly drill down on risk factors and regions and develop an initial risk assessment for applicants, and by streaming services to find viewers with similar behavior.

As the volume of text data has exploded at a rapid rate, it is imperative for organizations to have a framework in place to extract meaningful insights from the text being created. From social media analytics to risk management and cybercrime prevention, textual data management has never been more crucial. Cases where text clustering is significant include document retrieval, taxonomy development, spam mail filtering, and language translation. Clustering in text analysis is grouping a set of unlabeled texts so that texts within the same cluster are more similar to one another than those in other clusters. Text clustering algorithms analyze text and assess whether or not the data has natural cluster groups.

Methods

Data Preparation, Exploration, Visualization Process

For this project, 200 reviews of the following 20 movies were chosen from IMDB. Rotten Tomatoes, and other movie rating articles in order to efficiently construct a text corpus:

['Angel Has Fallen' 'Inception' 'No Time To Die' 'Taken' 'No Time To Die' 'Taxi' 'Despicable Me

3' 'Dirty Grandpa' 'Grown Ups' 'Legally Blonde' 'Lost City' 'Drag me to Hell' 'Fresh' 'It Chapter Two' 'The Toxic Avenger' 'Us' 'Batman' 'Everything Everywhere All at Once' 'Minority Report' 'Oblivion' 'Pitch Black']. Gensim, a free open-source Python library for topic modeling, document indexing, and similarity retrieval utilizing large corpora. This library is intended to handle raw, unstructured digital texts ("plain text") employing unsupervised machine learning techniques. Figure 1 depicts all of the other imported packages that were used.

```python
import pandas as pd
import os
import numpy as np
import re
import string
import seaborn as sns
import matplotlib.pyplot as plt
import nltk
import random
from dataclasses import dataclass

from nltk.corpus import stopwords
from nltk.stem.wordnet import WordNetLemmatizer
from nltk.stem import PorterStemmer

import gensim
from gensim import corpora, similarities
from gensim.models import Word2Vec, LdaMulticore, TfidfModel, CoherenceModel
from gensim.models.doc2vec import Doc2Vec, TaggedDocument
from gensim.models import LsiModel,LdaModel

from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.metrics.pairwise import cosine_similarity
from sklearn.manifold import TSNE, MDS
from sklearn.cluster import KMeans
from sklearn.metrics import roc_auc_score, accuracy_score, confusion_matrix, silhouette_score
from sklearn.svm import SVC
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score
from sklearn.model_selection import train_test_split, KFold
from sklearn.naive_bayes import MultinomialNB

from sklearn import metrics
from sklearn.metrics import confusion_matrix
from sklearn.metrics import f1_score

import scipy.cluster.hierarchy

from IPython.display import display, HTML

from typing import List, Callable, Dict
```

```
Genre of Movie  Movie Title
Action          Angel Has Fallen                     10
                Inception                            10
                No Time To Die                       10
                Taken                                10
                Taxi                                 10
Comedy          Despicable Me 3                      10
                Dirty Grandpa                        10
                Grown Ups                            10
                Legally Blonde                       10
                Lost City                            10
Horror          Drag me to Hell                      10
                Fresh                                10
                It Chapter Two                       10
                The Toxic Avenger                    10
                Us                                   10
Sci-Fi          Batman                               10
                Everything Everywhere All at Once    10
                Minority Report                      10
                Oblivion                             10
                Pitch Black                          10
Name: Review Type (pos or neg), dtype: int64
```

The corpus has 200 rows representing the movies reviews and 9 columns representing DSI_Title, Submission File Name, Student Name, Genre of Movie, Review Type (pos or neg), Movie Title, Text, Descriptor, and Doc_ID. The Figure 2 above shows number of reviews is 50 which is balanced across the four genres of movies: Action, Comedy, Horror, and Sci-fi. We continued to enhance the corpus-wide vocabulary and derived a few inferences based on classification, clustering, and topic modeling. Data wrangling included the following steps:

1. Removing punctuation
2. Removing English stopwords and a list of new words = ['movie', 'story', 'films']
3. Removing additional spaces and digits.
4. Lemmatization.
5. Retrieving the clean text.

Researching Design and Modeling Methods:

When categorizing objects, previously unseen items are placed into groups based on previously classified objects, often known as training data. This implies we have a solid baseline against which to compare new objects. Classification is therefore a supervised machine learning process. In order to determine whether or not two texts are 'similar', clustering methods compute a similarity or closeness measure, such as Euclidean distance. Clustering is an unsupervised strategy as all items are new upon clustering.

In general, document clustering may be accomplished by examining each document in vector format. Vectorization is nothing more than the development of a vocabulary vector. Word Embeddings, also known as Word Vectorization, is a technique used in natural language processing (NLP) to map words or phrases from a vocabulary to a vector of real numbers, which is then used to determine word similarities/semantics. Vectorization is the process of translating words into numbers. The following procedures were used such as Bag of words, which entails a series of steps, tokenization, vocabulary construction, and vector generation were utilized. Using a neural network, Word2vec will also learn distributed representations (word embeddings).

As a weighting factor, the TFIDF vectorizer was employed to turn words into vectors. By decreasing the impact of less significant words, this change increases the occurrence of rare terms. K-means was utilized to find underlying patterns and group comparable data points. Using the tf-idf and doc2vec methods, K-mean clustering was utilized. K-means assigns k random points in the vector space as the starting, virtual means of the k clusters, and then assigns each data point to the cluster mean closest to it. The actual mean of each cluster is then computed. The data points are reallocated based on the shift in the means. This procedure is repeated until the means of the clusters stop to move. To determine the appropriate number of clusters, the Silhouette Score was determined. Based on it, we selected the clusters that were sufficiently separated for a more in-depth examination. The Silhouette score ranges between -1 and 1. If the score is 1, the cluster is more dense and well-separated than others. A number close to 0 indicates overlapping clusters with samples extremely close to the decision border of neighboring clusters. A negative score [-1, 0] implies that the samples were maybe allocated to the incorrect clusters. We used various classification methods such as logistic regression, support vector classification, Naïve Bayes, and random forest and generated a few critical performance metrics such as confusion matrix, F1-score, and accuracy to evaluate the classification methods' performance for different vectorization methods.

Topic modeling algorithms are statistical approaches for analyzing the words of source texts in order to uncover the themes that run through them, how those topics are related to one another, and how they develop over time. To evaluate the documents and acquire greater insights for each approach, topic modeling was conducted using LSA and LDA model algorithms.

Researchers in psychology, information retrieval, and bibliometrics pioneered Latent Semantic Analysis (LSA), and it has been stated that this approach mimics how our human brain learns and forms conclusions. We utilized our database of movie reviews to explore how the reviews would be categorized using word-selection. LSA is fast easy to implement which is a single value decomposition SVD of rectangular Matrix this is similar to PCA. The only difference between PCA and SVD is that in PCA are we dealing with single correlation square matrices after that we extract eigenvectors, and eigenvalues. But SVD it’s a rectangular matrix we don’t have eigenvalue or eigenvectors but rather singular value or singular vectors. It does provide decent results which are much better vector space models.

The most well-known and widely utilized is Latent Dirichlet Allocation (LDA). Hidden variables are referred to as latent variables, a Dirichlet distribution is a probability distribution over other probability distributions, and distribution is the allocation of certain values based on the two. How should these topics be interpreted? A topic is defined as a probability distribution across words or documents. Some words or documents are more likely than others to appear in a topic. Documents are represented by LDA as a collection of topics. A topic is a collection of words. If a term is likely to appear in a topic, all documents containing that word will be more strongly connected with that topic as well. Furthermore, LDA indicates which subjects exist in each document and in what percentage. We conducted experiments with various combinations in order to offer recommendations and procedures for designing an effective corpus learning process. Text Classification is the automated process of categorizing text into predetermined groups. We divided the movie reviews into two categories: positive and negative. We used the following classification techniques after vectorization:

Logistic Regression: The weighted combination of the input features is used, and it is processed through a sigmoid function. The Sigmoid function converts any real number to a number between 0 and 1.

Support Vector Machine: Supervised learning models, together with accompanying learning algorithms, examine data for classification and regression analysis.

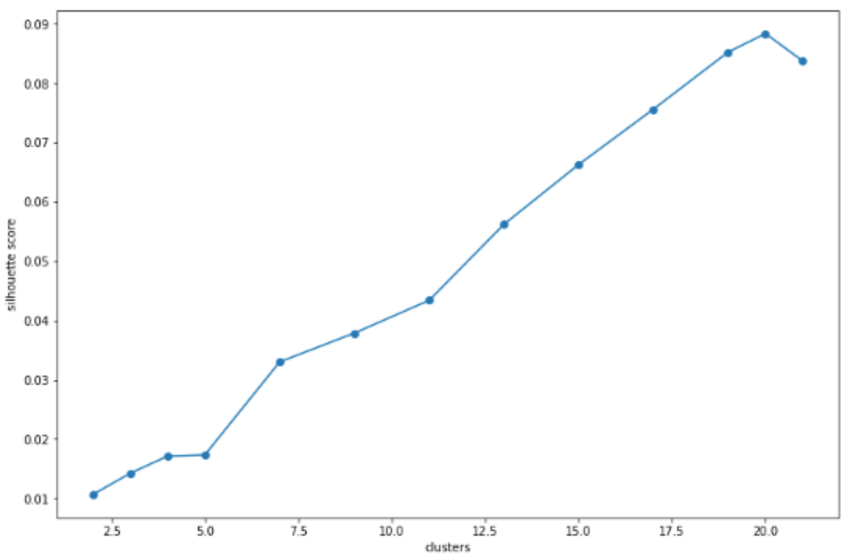
Naïve-Bayes: Naïve Bayes assigns probability to each positive and negative class for the provided documents.

Random Forest: At each phase, Gini/entropy are used as performance measures to differentiate between positive and negative classes with minimization.

The methods and techniques used aid in determining the ideal combination for clustering and classification in text analysis. This analysis is useful for determining what a document is about based on its keywords.

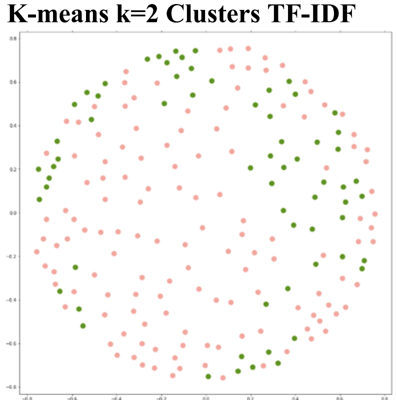
Results

We utilized the K-means technique to group similar data points together and find underlying patterns. We experimented with random states 5 and 10. However, because no significant difference was found, just the findings for random 5 are shown below. According to the silhouette scores, the best number of clusters is 20. We showed findings for 20 clusters as well as 2 and 4 clusters based on TF IDF.

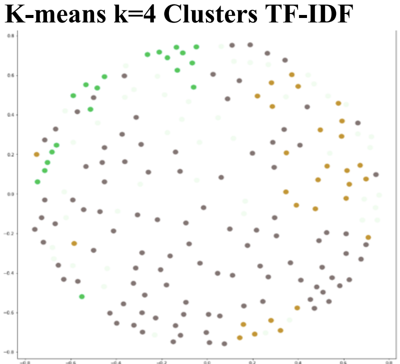



```
k=2, Silhouette Score: 0.011
k=3, Silhouette Score: 0.014
k=4, Silhouette Score: 0.017
k=5, Silhouette Score: 0.017
k=7, Silhouette Score: 0.033
k=9, Silhouette Score: 0.038
k=11, Silhouette Score: 0.043
k=13, Silhouette Score: 0.056
k=15, Silhouette Score: 0.066
k=17, Silhouette Score: 0.076
k=19, Silhouette Score: 0.085
k=20, Silhouette Score: 0.088
k=21, Silhouette Score: 0.084
```

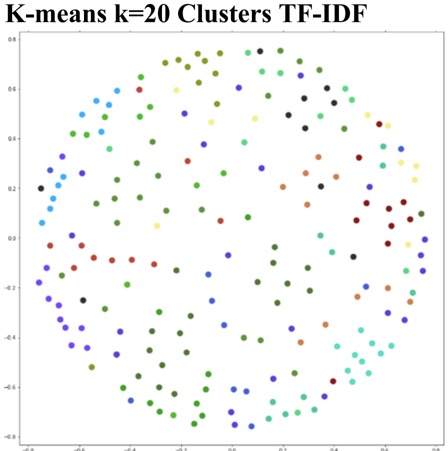
Movie Genres Color-Coding Legend: Action, Comedy, Horror, Sci-Fi

K-means k=2 Clusters TF-IDF  The findings shows that cluster 0 and segmentation only contains Action, and Sci-Fi genres. But cluster 1 is mixed of all Comedy, Action, Horror, and Sci-Fi indicating its not segmented as expected by experimenting k=2 clusters based on tf-idf matrix	0:['WCU_Doc1_NoTimeToDie', 'WCU_Doc5_NoTimeToDie', 'WCU_Doc7_NoTimeToDie', 'JGS_Doc2_Taken', 'JGS_Doc7_Taken', 'JGS_Doc9_Taken', 'DWH_Doc1_Inception', 'DWH_Doc2_Inception', 'DWH_Doc3_Inception', 'DWH_Doc4_Inception', 'DWH_Doc5_Inception', 'DWH_Doc6_Inception', 'DWH_Doc7_Inception', 'DWH_Doc8_Inception', 'DWH_Doc9_Inception', 'DWH_Doc10_Inception', 'LLQ_Doc1_AngelHasFallen', 'LLQ_Doc2_AngelHasFallen', 'LLQ_Doc3_AngelHasFallen', 'LLQ_Doc4_AngelHasFallen', 'LLQ_Doc5_AngelHasFallen', 'LLQ_Doc6_AngelHasFallen', 'LLQ_Doc7_AngelHasFallen', 'LLQ_Doc8_AngelHasFallen', 'LLQ_Doc9_AngelHasFallen', 'LLQ_Doc10_AngelHasFallen', 'XSP_Doc1_MinorityReport', 'XSP_Doc2_MinorityReport', 'XSP_Doc3_MinorityReport', 'XSP_Doc4_MinorityReport', 'XSP_Doc5_MinorityReport', 'XSP_Doc6_MinorityReport', 'XSP_Doc7_MinorityReport', 'XSP_Doc8_MinorityReport', 'XSP_Doc9_MinorityReport', 'XSP_Doc10_MinorityReport',

	'RTD_Doc1_Oblivion', 'RTD_Doc3_Oblivion', 'RTD_Doc4_Oblivion', 'RTD_Doc5_Oblivion', 'RTD_Doc6_Oblivion', 'RTD_Doc7_Oblivion', 'RTD_Doc8_Oblivion', 'RTD_Doc9_Oblivion', 'RTD_Doc10_Oblivion', 'YWM_Doc1_Batman', 'YWM_Doc2_Batman', 'YWM_Doc3_Batman', 'YWM_Doc4_Batman', 'YWM_Doc5_Batman', 'YWM_Doc6_Batman', 'YWM_Doc7_Batman', 'YWM_Doc8_Batman', 'YWM_Doc9_Batman', 'YWM_Doc10_Batman', 'TWH_Doc1_PitchBlack', 'TWH_Doc2_PitchBlack', 'TWH_Doc3_PitchBlack', 'TWH_Doc4_PitchBlack', 'TWH_Doc5_PitchBlack', 'TWH_Doc6_PitchBlack', 'TWH_Doc7_PitchBlack', 'TWH_Doc8_PitchBlack', 'TWH_Doc9_PitchBlack', 'TWH_Doc10_PitchBlack'], 1: ['CFA_Doc1_DespicableMe3', 'CFA_Doc2_DespicableMe3', 'CFA_Doc3_DespicableMe3', 'CFA_Doc4_DespicableMe3', 'CFA_Doc5_DespicableMe3', 'CFA_Doc6_DespicableMe3', 'CFA_Doc7_DespicableMe3', 'CFA_Doc8_DespicableMe3', 'CFA_Doc9_DespicableMe3', 'CFA_Doc10_DespicableMe3', 'ACB_Doc1_Grown Ups', 'ACB_Doc2_Grown Ups', 'ACB_Doc3_Grown Ups', 'ACB_Doc4_Grown Ups', 'ACB_Doc5_Grown Ups', 'ACB_Doc6_Grown Ups', 'ACB_Doc7_Grown Ups', 'ACB_Doc8_Grown Ups', 'ACB_Doc9_Grown Ups', 'ACB_Doc10_Grown Ups', 'VCT_Doc1_DirtyGrandpa', 'VCT_Doc2_DirtyGrandpa', 'VCT_Doc3_DirtyGrandpa', 'VCT_Doc4_DirtyGrandpa', 'VCT_Doc5_DirtyGrandpa', 'VCT_Doc6_DirtyGrandpa', 'VCT_Doc7_DirtyGrandpa', 'VCT_Doc8_DirtyGrandpa', 'VCT_Doc9_DirtyGrandpa', 'VCT_Doc10_DirtyGrandpa', 'HCC_DocW_LegallyBlonde1', 'HCC_DocW_LegallyBlonde2', 'HCC_DocW_LegallyBlonde3', 'HCC_DocW_LegallyBlonde4', 'HCC_DocW_LegallyBlonde5', 'HCC_DocW_LegallyBlonde6', 'HCC_DocW_LegallyBlonde7', 'HCC_DocW_LegallyBlonde8', 'HCC_DocW_LegallyBlonde9', 'HCC_DocW_LegallyBlonde10', 'GCS_Doc1_LostCity', 'GCS_Doc2_LostCity', 'GCS_Doc3_LostCity', 'GCS_Doc4_LostCity', 'GCS_Doc5_LostCity', 'GCS_Doc6_LostCity', 'GCS_Doc7_LostCity', 'GCS_Doc8_LostCity', 'GCS_Doc9_LostCity', 'GCS_Doc10_LostCity', 'WCU_Doc2_NoTimeToDie', 'WCU_Doc3_NoTimeToDie', 'WCU_Doc4_NoTimeToDie', 'WCU_Doc6_NoTimeToDie', 'WCU_Doc8_NoTimeToDie', 'WCU_Doc9_NoTimeToDie', 'WCU_Doc10_NoTimeToDie', 'MDP_Doc1_Taxi', 'MDP_Doc2_Taxi', 'MDP_Doc3_Taxi', 'MDP_Doc4_Taxi', 'MDP_Doc5_Taxi', 'MDP_Doc6_Taxi', 'MDP_Doc7_Taxi', 'MDP_Doc8_Taxi', 'MDP_Doc9_Taxi', 'MDP_Doc10_Taxi', 'JGS_Doc1_Taken', 'JGS_Doc3_Taken', 'JGS_Doc4_Taken', 'JGS_Doc5_Taken', 'JGS_Doc6_Taken', 'JGS_Doc8_Taken',

	'JGS_Doc10_Taken', 'KCM_Doc1_ItChapterTwo', 'KCM_Doc2_ItChapterTwo', 'KCM_Doc3_ItChapterTwo', 'KCM_Doc4_ItChapterTwo', 'KCM_Doc5_ItChapterTwo', 'KCM_Doc6_ItChapterTwo', 'KCM_Doc7_ItChapterTwo', 'KCM_Doc8_ItChapterTwo', 'KCM_Doc9_ItChapterTwo', 'KCM_Doc10_ItChapterTwo', 'AJN_Doc1_TheToxicAvenger', 'AJN_Doc2_TheToxicAvenger', 'AJN_Doc3_TheToxicAvenger', 'AJN_Doc4_TheToxicAvenger', 'AJN_Doc5_TheToxicAvenger', 'AJN_Doc6_TheToxicAvenger', 'AJN_Doc7_TheToxicAvenger', 'AJN_Doc8_TheToxicAvenger', 'AJN_Doc9_TheToxicAvenger', 'AJN_Doc10_TheToxicAvenger', 'DSP_Doc1_Us', 'DSP_Doc2_Us', 'DSP_Doc3_Us', 'DSP_Doc4_Us', 'DSP_Doc5_Us', 'DSP_Doc6_Us', 'DSP_Doc7_Us', 'DSP_Doc8_Us', 'DSP_Doc9_Us', 'DSP_Doc10_Us', 'NPO_Doc1_DragMeToHell', 'NPO_Doc2_DragMeToHell', 'NPO_Doc3_DragMeToHell', 'NPO_Doc4_DragMeToHell', 'NPO_Doc5_DragMeToHell', 'NPO_Doc6_DragMeToHell', 'NPO_Doc7_DragMeToHell', 'NPO_Doc8_DragMeToHell', 'NPO_Doc9_DragMeToHell', 'NPO_Doc10_DragMeToHell', 'TRH_Doc1_Fresh', 'TRH_Doc2_Fresh', 'TRH_Doc3_Fresh', 'TRH_Doc4_Fresh', 'TRH_Doc5_Fresh', 'TRH_Doc6_Fresh', 'TRH_Doc7_Fresh', 'TRH_Doc8_Fresh', 'TRH_Doc9_Fresh', 'TRH_Doc10_Fresh', 'KST_Doc1_EEAaO', 'KST_Doc2_EEAaO', 'KST_Doc4_EEAaO', 'KST_Doc8_EEAaO', 'KST_Doc10_EEAaO', 'KST_Doc3_EEAaO', 'KST_Doc5_EEAaO', 'KST_Doc6_EEAaO', 'KST_Doc7_EEAaO', 'KST_Doc9_EEAaO', 'RTD_Doc2_Oblivion']
K-means k=4 Clusters TF-IDF  The findings shows that cluster 0 and 4 are not segmented properly since it contains a mix of all genres. Similarly, cluster 1 contains both Action and Sci-Fi movies genres. Based on our observation only cluster 2 segmented properly it has only Sci-Fi genre movie review as expected by experiment k=4 clusters based on tf-idf matrix	0: ['CFA_Doc1_DespicableMe3', 'CFA_Doc4_DespicableMe3', 'CFA_Doc5_DespicableMe3', 'CFA_Doc7_DespicableMe3', 'CFA_Doc8_DespicableMe3', 'CFA_Doc9_DespicableMe3', 'CFA_Doc10_DespicableMe3', 'ACB_Doc1_Grown Ups', 'ACB_Doc2_Grown Ups', 'ACB_Doc3_Grown Ups', 'ACB_Doc4_Grown Ups', 'ACB_Doc5_Grown Ups', 'ACB_Doc6_Grown Ups', 'ACB_Doc7_Grown Ups', 'ACB_Doc8_Grown Ups', 'ACB_Doc9_Grown Ups', 'ACB_Doc10_Grown Ups', 'VCT_Doc1_DirtyGrandpa', 'VCT_Doc2_DirtyGrandpa', 'VCT_Doc3_DirtyGrandpa', 'VCT_Doc4_DirtyGrandpa', 'VCT_Doc5_DirtyGrandpa', 'VCT_Doc6_DirtyGrandpa', 'VCT_Doc7_DirtyGrandpa', 'VCT_Doc8_DirtyGrandpa', 'VCT_Doc9_DirtyGrandpa', 'VCT_Doc10_DirtyGrandpa', 'HCC_DocW_LegallyBlonde1', 'HCC_DocW_LegallyBlonde2', 'HCC_DocW_LegallyBlonde3', 'HCC_DocW_LegallyBlonde4', 'HCC_DocW_LegallyBlonde5', 'HCC_DocW_LegallyBlonde6', 'HCC_DocW_LegallyBlonde7', 'HCC_DocW_LegallyBlonde8', 'HCC_DocW_LegallyBlonde9', 'HCC_DocW_LegallyBlonde10', 'GCS_Doc1_LostCity', 'GCS_Doc2_LostCity', 'GCS_Doc3_LostCity',

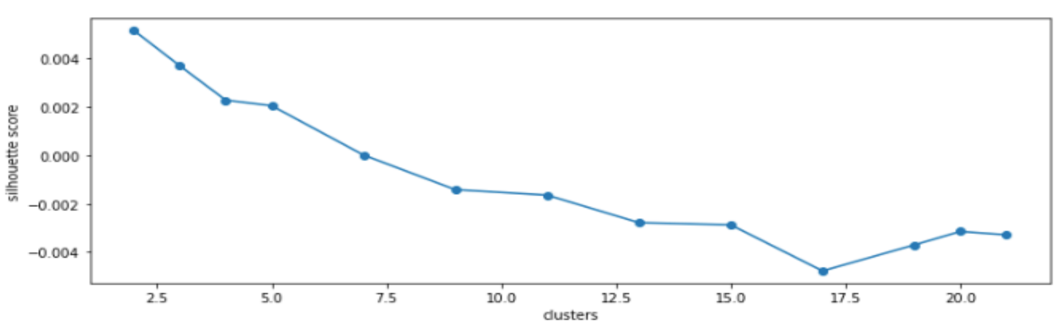
	'GCS_Doc4_LostCity', 'GCS_Doc5_LostCity', 'GCS_Doc6_LostCity', 'GCS_Doc7_LostCity', 'GCS_Doc8_LostCity', 'GCS_Doc9_LostCity', 'GCS_Doc10_LostCity', 'WCU_Doc3_NoTimeToDie', 'WCU_Doc4_NoTimeToDie', 'WCU_Doc5_NoTimeToDie', 'WCU_Doc6_NoTimeToDie', 'WCU_Doc7_NoTimeToDie', 'WCU_Doc8_NoTimeToDie', 'WCU_Doc10_NoTimeToDie', 'MDP_Doc1_Taxi', 'MDP_Doc2_Taxi', 'MDP_Doc3_Taxi', 'MDP_Doc4_Taxi', 'MDP_Doc5_Taxi', 'MDP_Doc6_Taxi', 'MDP_Doc7_Taxi', 'MDP_Doc8_Taxi', 'MDP_Doc9_Taxi', 'MDP_Doc10_Taxi', 'JGS_Doc1_Taken', 'JGS_Doc2_Taken', 'JGS_Doc3_Taken', 'JGS_Doc6_Taken', 'JGS_Doc7_Taken', 'JGS_Doc9_Taken', 'JGS_Doc10_Taken', 'LLQ_Doc1_AngelHasFallen', 'LLQ_Doc2_AngelHasFallen', 'LLQ_Doc3_AngelHasFallen', 'LLQ_Doc4_AngelHasFallen', 'LLQ_Doc5_AngelHasFallen', 'LLQ_Doc6_AngelHasFallen', 'LLQ_Doc7_AngelHasFallen', 'LLQ_Doc8_AngelHasFallen', 'LLQ_Doc9_AngelHasFallen', 'LLQ_Doc10_AngelHasFallen', 'KCM_Doc1_ItChapterTwo', 'KCM_Doc4_ItChapterTwo', 'KCM_Doc5_ItChapterTwo', 'KCM_Doc6_ItChapterTwo', 'KCM_Doc8_ItChapterTwo', 'KCM_Doc10_ItChapterTwo', 'AJN_Doc1_TheToxicAvenger', 'AJN_Doc2_TheToxicAvenger', 'AJN_Doc3_TheToxicAvenger', 'AJN_Doc4_TheToxicAvenger', 'AJN_Doc5_TheToxicAvenger', 'AJN_Doc6_TheToxicAvenger', 'AJN_Doc7_TheToxicAvenger', 'AJN_Doc8_TheToxicAvenger', 'AJN_Doc9_TheToxicAvenger', 'AJN_Doc10_TheToxicAvenger', 'NPO_Doc7_DragMeToHell', 'TRH_Doc1_Fresh', 'TRH_Doc4_Fresh', 'TRH_Doc8_Fresh', 'TRH_Doc9_Fresh', 'RTD_Doc2_Oblivion'], 1: ['DWH_Doc1_Inception', 'DWH_Doc2_Inception', 'DWH_Doc3_Inception', 'DWH_Doc4_Inception', 'DWH_Doc5_Inception', 'DWH_Doc6_Inception', 'DWH_Doc7_Inception', 'DWH_Doc8_Inception', 'DWH_Doc9_Inception', 'DWH_Doc10_Inception', 'YWM_Doc1_Batman', 'YWM_Doc2_Batman', 'YWM_Doc3_Batman', 'YWM_Doc4_Batman', 'YWM_Doc5_Batman', 'YWM_Doc6_Batman', 'YWM_Doc7_Batman', 'YWM_Doc8_Batman', 'YWM_Doc9_Batman', 'YWM_Doc10_Batman'], 2: ['XSP_Doc1_MinorityReport', 'XSP_Doc2_MinorityReport', 'XSP_Doc3_MinorityReport', 'XSP_Doc4_MinorityReport', 'XSP_Doc5_MinorityReport', 'XSP_Doc6_MinorityReport', 'XSP_Doc7_MinorityReport', 'XSP_Doc8_MinorityReport', 'XSP_Doc9_MinorityReport', 'XSP_Doc10_MinorityReport', 'RTD_Doc1_Oblivion', 'RTD_Doc3_Oblivion', 'RTD_Doc4_Oblivion', 'RTD_Doc5_Oblivion', 'RTD_Doc6_Oblivion', 'RTD_Doc7_Oblivion', 'RTD_Doc8_Oblivion', 'RTD_Doc9_Oblivion', 'RTD_Doc10_Oblivion',

	'TWH_Doc1_PitchBlack', 'TWH_Doc2_PitchBlack', 'TWH_Doc3_PitchBlack', 'TWH_Doc4_PitchBlack', 'TWH_Doc5_PitchBlack', 'TWH_Doc6_PitchBlack', 'TWH_Doc7_PitchBlack', 'TWH_Doc8_PitchBlack', 'TWH_Doc9_PitchBlack', 'TWH_Doc10_PitchBlack'], 3: ['CFA_Doc2_DespicableMe3', 'CFA_Doc3_DespicableMe3', 'CFA_Doc6_DespicableMe3', 'WCU_Doc1_NoTimeToDie', 'WCU_Doc2_NoTimeToDie', 'WCU_Doc9_NoTimeToDie', 'JGS_Doc4_Taken', 'JGS_Doc5_Taken', 'JGS_Doc8_Taken', 'KCM_Doc2_ItChapterTwo', 'KCM_Doc3_ItChapterTwo', 'KCM_Doc7_ItChapterTwo', 'KCM_Doc9_ItChapterTwo', 'DSP_Doc1_Us', 'DSP_Doc2_Us', 'DSP_Doc3_Us', 'DSP_Doc4_Us', 'DSP_Doc5_Us', 'DSP_Doc6_Us', 'DSP_Doc7_Us', 'DSP_Doc8_Us', 'DSP_Doc9_Us', 'DSP_Doc10_Us', 'NPO_Doc1_DragMeToHell', 'NPO_Doc2_DragMeToHell', 'NPO_Doc3_DragMeToHell', 'NPO_Doc4_DragMeToHell', 'NPO_Doc5_DragMeToHell', 'NPO_Doc6_DragMeToHell', 'NPO_Doc8_DragMeToHell', 'NPO_Doc9_DragMeToHell', 'NPO_Doc10_DragMeToHell', 'TRH_Doc2_Fresh', 'TRH_Doc3_Fresh', 'TRH_Doc5_Fresh', 'TRH_Doc6_Fresh', 'TRH_Doc7_Fresh', 'TRH_Doc10_Fresh', 'KST_Doc1_EEAaO', 'KST_Doc2_EEAaO', 'KST_Doc4_EEAaO', 'KST_Doc8_EEAaO', 'KST_Doc10_EEAaO', 'KST_Doc3_EEAaO', 'KST_Doc5_EEAaO', 'KST_Doc6_EEAaO', 'KST_Doc7_EEAaO', 'KST_Doc9_EEAaO']
K-means k=20 Clusters TF-IDF  The findings show that all the clusters are segmented properly as expected since does not contain a mix of genres. These are the results of experimenting with k=20 clusters based on the tf-idf matrix which performed really well compared to other experiments.	0: ['TRH_Doc1_Fresh', 'TRH_Doc2_Fresh', 'TRH_Doc3_Fresh', 'TRH_Doc4_Fresh', 'TRH_Doc5_Fresh', 'TRH_Doc6_Fresh', 'TRH_Doc7_Fresh', 'TRH_Doc8_Fresh', 'TRH_Doc9_Fresh', 'TRH_Doc10_Fresh'], 1: ['JGS_Doc1_Taken', 'JGS_Doc2_Taken', 'JGS_Doc3_Taken', 'JGS_Doc4_Taken', 'JGS_Doc5_Taken', 'JGS_Doc6_Taken', 'JGS_Doc7_Taken', 'JGS_Doc8_Taken', 'JGS_Doc9_Taken', 'JGS_Doc10_Taken'], 2: ['GCS_Doc1_LostCity', 'GCS_Doc2_LostCity', 'GCS_Doc3_LostCity', 'GCS_Doc4_LostCity', 'GCS_Doc5_LostCity', 'GCS_Doc6_LostCity', 'GCS_Doc7_LostCity', 'GCS_Doc8_LostCity', 'GCS_Doc9_LostCity', 'GCS_Doc10_LostCity'], 3: ['YWM_Doc1_Batman', 'YWM_Doc2_Batman', 'YWM_Doc3_Batman', 'YWM_Doc4_Batman', 'YWM_Doc5_Batman', 'YWM_Doc6_Batman', 'YWM_Doc7_Batman', 'YWM_Doc8_Batman', 'YWM_Doc9_Batman', 'YWM_Doc10_Batman'],

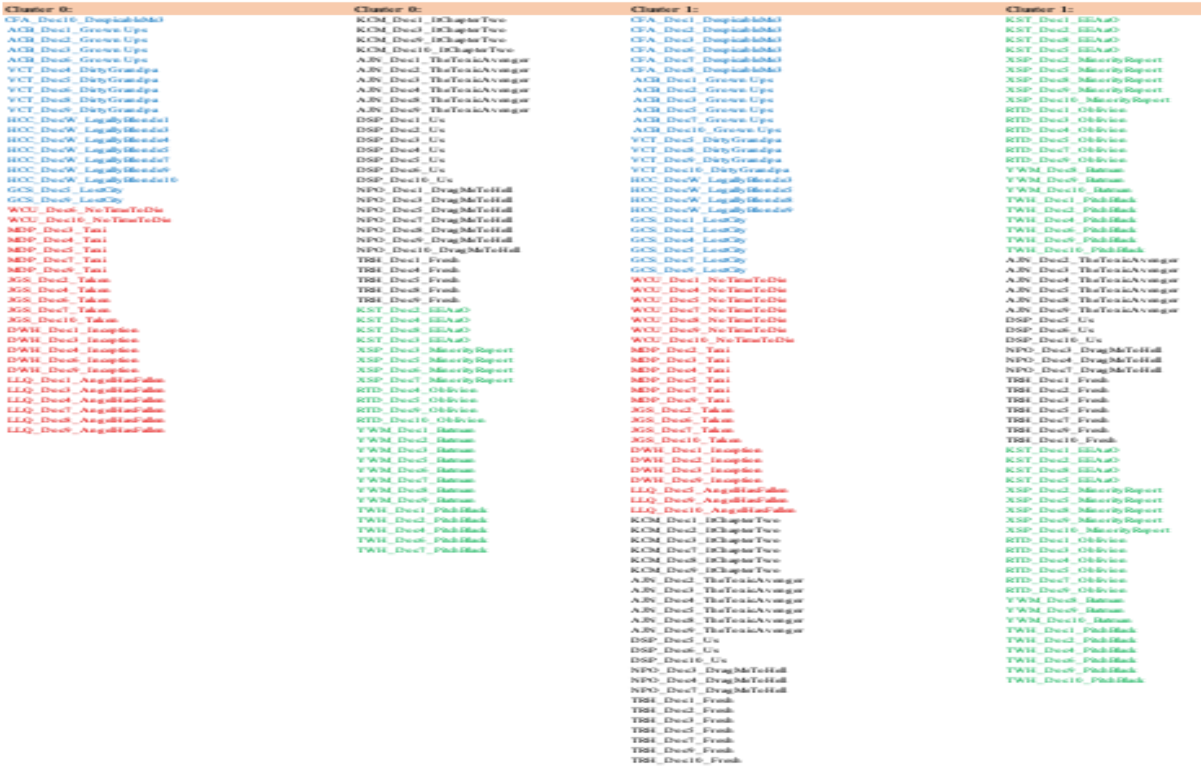
	4: ['DWH_Doc1_Inception', 'DWH_Doc2_Inception', 'DWH_Doc3_Inception', 'DWH_Doc4_Inception', 'DWH_Doc5_Inception', 'DWH_Doc6_Inception', 'DWH_Doc7_Inception', 'DWH_Doc8_Inception', 'DWH_Doc9_Inception', 'DWH_Doc10_Inception'], 5: ['KST_Doc1_EEAaO', 'KST_Doc2_EEAaO', 'KST_Doc4_EEAaO', 'KST_Doc8_EEAaO', 'KST_Doc10_EEAaO', 'KST_Doc3_EEAaO', 'KST_Doc5_EEAaO', 'KST_Doc6_EEAaO', 'KST_Doc7_EEAaO', 'KST_Doc9_EEAaO'], 6: ['TWH_Doc1_PitchBlack', 'TWH_Doc2_PitchBlack', 'TWH_Doc3_PitchBlack', 'TWH_Doc4_PitchBlack', 'TWH_Doc5_PitchBlack', 'TWH_Doc6_PitchBlack', 'TWH_Doc7_PitchBlack', 'TWH_Doc8_PitchBlack', 'TWH_Doc9_PitchBlack', 'TWH_Doc10_PitchBlack'], 7: ['HCC_DocW_LegallyBlonde1', 'HCC_DocW_LegallyBlonde2', 'HCC_DocW_LegallyBlonde3', 'HCC_DocW_LegallyBlonde4', 'HCC_DocW_LegallyBlonde5', 'HCC_DocW_LegallyBlonde6', 'HCC_DocW_LegallyBlonde7', 'HCC_DocW_LegallyBlonde8', 'HCC_DocW_LegallyBlonde9', 'HCC_DocW_LegallyBlonde10'], 8: ['KCM_Doc1_ItChapterTwo', 'KCM_Doc2_ItChapterTwo', 'KCM_Doc3_ItChapterTwo', 'KCM_Doc4_ItChapterTwo', 'KCM_Doc5_ItChapterTwo', 'KCM_Doc6_ItChapterTwo', 'KCM_Doc7_ItChapterTwo', 'KCM_Doc8_ItChapterTwo', 'KCM_Doc9_ItChapterTwo', 'KCM_Doc10_ItChapterTwo'], 9: ['CFA_Doc1_DespicableMe3', 'CFA_Doc2_DespicableMe3', 'CFA_Doc3_DespicableMe3', 'CFA_Doc4_DespicableMe3', 'CFA_Doc5_DespicableMe3', 'CFA_Doc6_DespicableMe3', 'CFA_Doc7_DespicableMe3', 'CFA_Doc8_DespicableMe3', 'CFA_Doc9_DespicableMe3', 'CFA_Doc10_DespicableMe3'], 10: ['DSP_Doc1_Us', 'DSP_Doc2_Us', 'DSP_Doc3_Us', 'DSP_Doc4_Us', 'DSP_Doc5_Us', 'DSP_Doc6_Us', 'DSP_Doc7_Us', 'DSP_Doc8_Us', 'DSP_Doc9_Us', 'DSP_Doc10_Us'], 11: ['AJN_Doc1_TheToxicAvenger', 'AJN_Doc2_TheToxicAvenger', 'AJN_Doc3_TheToxicAvenger', 'AJN_Doc4_TheToxicAvenger', 'AJN_Doc5_TheToxicAvenger', 'AJN_Doc6_TheToxicAvenger', 'AJN_Doc7_TheToxicAvenger', 'AJN_Doc8_TheToxicAvenger', 'AJN_Doc9_TheToxicAvenger', 'AJN_Doc10_TheToxicAvenger'], 12: ['XSP_Doc1_MinorityReport', 'XSP_Doc2_MinorityReport', 'XSP_Doc3_MinorityReport', 'XSP_Doc4_MinorityReport', 'XSP_Doc5_MinorityReport', 'XSP_Doc6_MinorityReport', 'XSP_Doc7_MinorityReport', 'XSP_Doc8_MinorityReport', 'XSP_Doc9_MinorityReport', 'XSP_Doc10_MinorityReport'], 13: ['RTD_Doc1_Oblivion', 'RTD_Doc2_Oblivion',

	'RTD_Doc3_Oblivion', 'RTD_Doc4_Oblivion', 'RTD_Doc5_Oblivion', 'RTD_Doc6_Oblivion', 'RTD_Doc7_Oblivion', 'RTD_Doc8_Oblivion', 'RTD_Doc9_Oblivion', 'RTD_Doc10_Oblivion'], 14: ['VCT_Doc1_DirtyGrandpa', 'VCT_Doc2_DirtyGrandpa', 'VCT_Doc3_DirtyGrandpa', 'VCT_Doc4_DirtyGrandpa', 'VCT_Doc5_DirtyGrandpa', 'VCT_Doc6_DirtyGrandpa', 'VCT_Doc7_DirtyGrandpa', 'VCT_Doc8_DirtyGrandpa', 'VCT_Doc9_DirtyGrandpa', 'VCT_Doc10_DirtyGrandpa'], 15: ['MDP_Doc1_Taxi', 'MDP_Doc2_Taxi', 'MDP_Doc3_Taxi', 'MDP_Doc4_Taxi', 'MDP_Doc5_Taxi', 'MDP_Doc6_Taxi', 'MDP_Doc7_Taxi', 'MDP_Doc8_Taxi', 'MDP_Doc9_Taxi', 'MDP_Doc10_Taxi'], 16: ['WCU_Doc1_NoTimeToDie', 'WCU_Doc2_NoTimeToDie', 'WCU_Doc3_NoTimeToDie', 'WCU_Doc4_NoTimeToDie', 'WCU_Doc5_NoTimeToDie', 'WCU_Doc6_NoTimeToDie', 'WCU_Doc7_NoTimeToDie', 'WCU_Doc8_NoTimeToDie', 'WCU_Doc9_NoTimeToDie', 'WCU_Doc10_NoTimeToDie'], 17: ['ACB_Doc1_Grown Ups', 'ACB_Doc2_Grown Ups', 'ACB_Doc3_Grown Ups', 'ACB_Doc4_Grown Ups', 'ACB_Doc5_Grown Ups', 'ACB_Doc6_Grown Ups', 'ACB_Doc7_Grown Ups', 'ACB_Doc8_Grown Ups', 'ACB_Doc9_Grown Ups', 'ACB_Doc10_Grown Ups'], 18: ['NPO_Doc1_DragMeToHell', 'NPO_Doc2_DragMeToHell', 'NPO_Doc3_DragMeToHell', 'NPO_Doc4_DragMeToHell', 'NPO_Doc5_DragMeToHell', 'NPO_Doc6_DragMeToHell', 'NPO_Doc7_DragMeToHell', 'NPO_Doc8_DragMeToHell', 'NPO_Doc9_DragMeToHell', 'NPO_Doc10_DragMeToHell'], 19: ['LLQ_Doc1_AngelHasFallen', 'LLQ_Doc2_AngelHasFallen', 'LLQ_Doc3_AngelHasFallen', 'LLQ_Doc4_AngelHasFallen', 'LLQ_Doc5_AngelHasFallen', 'LLQ_Doc6_AngelHasFallen', 'LLQ_Doc7_AngelHasFallen', 'LLQ_Doc8_AngelHasFallen', 'LLQ_Doc9_AngelHasFallen', 'LLQ_Doc10_AngelHasFallen']

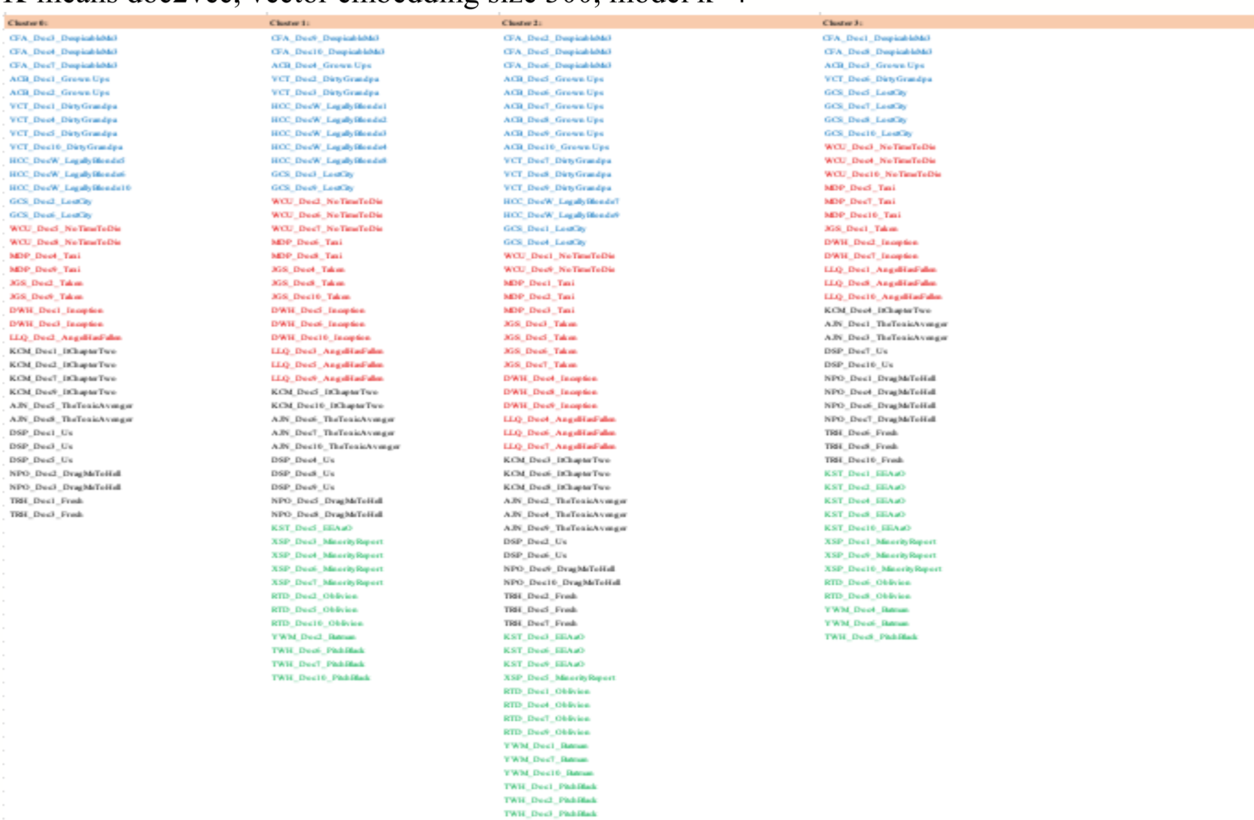
Based on the silhouette scores obtained, we also compared K-means clusters for 2, 4, 13, and 20 clusters for doc2vec, vector embedding size 300 and 600 and obtained the following results:



K-means doc2vec, vector embedding size 300, model k=2



K-means doc2vec, vector embedding size 300, model k=4



K-means doc2vec model, vector embedding size 300, k=13

Cluster 0:	Cluster 1:	Cluster 2:	Cluster 3:	Cluster 4:	Cluster 5:	Cluster 6:
ACB_Doc2_Grown Ups	VCT_Doc9_DirtyGrandpa	CFA_Doc5_DespicableMe3	CFA_Doc10_DespicableMe3	VCT_Doc5_DirtyGrandpa	DWH_Doc7_Inception	CFA_Doc1_DespicableMe3
ACB_Doc7_Grown Ups	HCC_DocW_LegallyBlonde3	CFA_Doc9_DespicableMe3	DSP_Doc8_Us	HCC_DocW_LegallyBlonde8	LLQ_Doc7_AngelHasFallen	CFA_Doc8_DespicableMe3
VCT_Doc8_DirtyGrandpa	HCC_DocW_LegallyBlonde6	ACB_Doc5_Grown Ups	NPO_Doc5_DragMeToHell	GCS_Doc3_LostCity	KCM_Doc4_ItChapterTwo	ACB_Doc1_Grown Ups
GCS_Doc4_LostCity	HCC_DocW_LegallyBlonde10	HCC_DocW_LegallyBlonde2	TRH_Doc1_Fresh	GCS_Doc8_LostCity	DSP_Doc7_Us	VCT_Doc7_DirtyGrandpa
GCS_Doc6_LostCity	WCU_Doc7_NoTimeToDie	HCC_DocW_LegallyBlonde4	XSP_Doc1_MinorityReport	TRH_Doc6_Fresh	TWH_Doc1_PitchBlack	WCU_Doc1_NoTimeToDie
KCM_Doc3_ItChapterTwo	JGS_Doc7_Taken	MDP_Doc4_Taxi		RTD_Doc10_Oblivion	TWH_Doc8_PitchBlack	WCU_Doc8_NoTimeToDie
KCM_Doc8_ItChapterTwo	AJN_Doc1_TheToxicAvenger	JGS_Doc2_Taken				JGS_Doc10_Taken
AJN_Doc2_TheToxicAvenger	AJN_Doc4_TheToxicAvenger	JGS_Doc5_Taken				DWH_Doc1_Inception
	DSP_Doc3_Us	LLQ_Doc4_AngelHasFallen				AJN_Doc7_TheToxicAvenger
	KST_Doc4_EEAaO	TRH_Doc4_Fresh				DSP_Doc4_Us
	XSP_Doc3_MinorityReport	KST_Doc6_EEAaO				DSP_Doc5_Us
	XSP_Doc9_MinorityReport	RTD_Doc1_Oblivion				KST_Doc8_EEAaO
	TWH_Doc4_PitchBlack	RTD_Doc7_Oblivion				KST_Doc5_EEAaO
						XSP_Doc6_MinorityReport
						XSP_Doc10_MinorityReport
						YWM_Doc5_Batman

Cluster 7:	Cluster 8:	Cluster 9:	Cluster 10:	Cluster 11:	Cluster 12:
ACB_Doc6_Grown Ups	CFA_Doc2_DespicableMe3	VCT_Doc2_DirtyGrandpa	ACB_Doc8_Grown Ups	CFA_Doc3_DespicableMe3	ACB_Doc3_Grown Ups
VCT_Doc1_DirtyGrandpa	CFA_Doc4_DespicableMe3	WCU_Doc3_NoTimeToDie	GCS_Doc2_LostCity	CFA_Doc7_DespicableMe3	ACB_Doc9_Grown Ups
VCT_Doc10_DirtyGrandpa	CFA_Doc6_DespicableMe3	JGS_Doc6_Taken	GCS_Doc5_LostCity	ACB_Doc4_Grown Ups	ACB_Doc10_Grown Ups
HCC_DocW_LegallyBlonde7	VCT_Doc4_DirtyGrandpa	DWH_Doc3_Inception	WCU_Doc2_NoTimeToDie	GCS_Doc1_LostCity	VCT_Doc3_DirtyGrandpa
GCS_Doc7_LostCity	VCT_Doc6_DirtyGrandpa	DSP_Doc9_Us	MDP_Doc5_Taxi	GCS_Doc9_LostCity	WCU_Doc9_NoTimeToDie
WCU_Doc6_NoTimeToDie	HCC_DocW_LegallyBlonde1	NPO_Doc4_DragMeToHell	MDP_Doc6_Taxi	WCU_Doc4_NoTimeToDie	MDP_Doc9_Taxi
MDP_Doc8_Taxi	HCC_DocW_LegallyBlonde5	KST_Doc9_EEAaO	MDP_Doc7_Taxi	WCU_Doc5_NoTimeToDie	JGS_Doc1_Taken
MDP_Doc10_Taxi	HCC_DocW_LegallyBlonde9	TWH_Doc7_PitchBlack	JGS_Doc8_Taken	WCU_Doc10_NoTimeToDie	LLQ_Doc3_AngelHasFallen
JGS_Doc3_Taken	GCS_Doc10_LostCity	TWH_Doc10_PitchBlack	DWH_Doc8_Inception	MDP_Doc3_Taxi	LLQ_Doc9_AngelHasFallen
DWH_Doc4_Inception	MDP_Doc1_Taxi		LLQ_Doc2_AngelHasFallen	JGS_Doc9_Taken	KCM_Doc1_ItChapterTwo
DWH_Doc5_Inception	MDP_Doc2_Taxi		KCM_Doc5_ItChapterTwo	DWH_Doc6_Inception	AJN_Doc5_TheToxicAvenger
DWH_Doc10_Inception	JGS_Doc4_Taken		DSP_Doc2_Us	AJN_Doc6_TheToxicAvenger	RTD_Doc9_Oblivion
LLQ_Doc1_AngelHasFallen	DWH_Doc2_Inception		KST_Doc2_EEAaO	AJN_Doc10_TheToxicAvenger	YWM_Doc2_Batman
LLQ_Doc6_AngelHasFallen	DWH_Doc9_Inception		XSP_Doc5_MinorityReport	NPO_Doc3_DragMeToHell	YWM_Doc8_Batman
LLQ_Doc8_AngelHasFallen	LLQ_Doc5_AngelHasFallen			NPO_Doc7_DragMeToHell	
LLQ_Doc10_AngelHasFallen	KCM_Doc9_ItChapterTwo			TRH_Doc2_Fresh	
KCM_Doc2_ItChapterTwo	AJN_Doc8_TheToxicAvenger			KST_Doc1_EEAaO	
KCM_Doc6_ItChapterTwo	DSP_Doc1_Us				
KCM_Doc7_ItChapterTwo	DSP_Doc10_Us				
KCM_Doc10_ItChapterTwo	NPO_Doc8_DragMeToHell				
AJN_Doc3_TheToxicAvenger	NPO_Doc9_DragMeToHell				
AJN_Doc9_TheToxicAvenger	NPO_Doc10_DragMeToHell				

K-means doc2vec model, vector embedding size 300, k =20

Cluster 0:	Cluster 1:	Cluster 2:	Cluster 3:	Cluster 4:	Cluster 5:	Cluster 6:
HCC_DocW_LegallyBlonde10	CFA_Doc6_DespicableMe3	JGS_Doc8_Taken	ACB_Doc9_Grown Ups	VCT_Doc5_DirtyGrandpa	CFA_Doc1_DespicableMe3	CFA_Doc7_DespicableMe3
WCU_Doc7_NoTimeToDie	HCC_DocW_LegallyBlonde8	DWH_Doc3_Inception	DWH_Doc2_Inception	HCC_DocW_LegallyBlonde7	CFA_Doc2_DespicableMe3	ACB_Doc5_Grown Ups
MDP_Doc7_Taxi	GCS_Doc5_LostCity	LLQ_Doc6_AngelHasFallen	AJN_Doc2_TheToxicAvenger	MDP_Doc8_Taxi	CFA_Doc8_DespicableMe3	JGS_Doc7_Taken
JGS_Doc6_Taken	MDP_Doc2_Taxi	KCM_Doc6_ItChapterTwo	TRH_Doc2_Fresh	LLQ_Doc8_AngelHasFallen	ACB_Doc1_Grown Ups	AJN_Doc8_TheToxicAvenger
AJN_Doc4_TheToxicAvenger	MDP_Doc9_Taxi	DSP_Doc1_Us	XSP_Doc5_MinorityReport	AJN_Doc3_TheToxicAvenger	VCT_Doc7_DirtyGrandpa	NPO_Doc4_DragMeToHell
TRH_Doc8_Fresh	TRH_Doc10_Fresh	DSP_Doc10_Us	RTD_Doc9_Oblivion	AJN_Doc6_TheToxicAvenger	GCS_Doc2_LostCity	KST_Doc4_EEAaO
TWH_Doc1_PitchBlack	RTD_Doc1_Oblivion	TRH_Doc3_Fresh	YWM_Doc1_Batman	AJN_Doc10_TheToxicAvenger	WCU_Doc1_NoTimeToDie	KST_Doc6_EEAaO
TWH_Doc2_PitchBlack		KST_Doc2_EEAaO	TWH_Doc3_PitchBlack	NPO_Doc1_DragMeToHell	WCU_Doc8_NoTimeToDie	XSP_Doc9_MinorityReport
TWH_Doc7_PitchBlack		KST_Doc9_EEAaO		NPO_Doc6_DragMeToHell	JGS_Doc10_Taken	XSP_Doc10_MinorityReport
		TWH_Doc10_PitchBlack		TRH_Doc6_Fresh	DWH_Doc1_Inception	
				XSP_Doc1_MinorityReport	LLQ_Doc2_AngelHasFallen	
				RTD_Doc3_Oblivion	LLQ_Doc10_AngelHasFallen	
					AJN_Doc7_TheToxicAvenger	
					DSP_Doc3_Us	
					DSP_Doc4_Us	
					DSP_Doc5_Us	
					TRH_Doc7_Fresh	
					KST_Doc8_EEAaO	
					KST_Doc5_EEAaO	
					XSP_Doc6_MinorityReport	
					YWM_Doc3_Batman	
					YWM_Doc5_Batman	
					YWM_Doc9_Batman	

Cluster 7:	Cluster 8:	Cluster 9:	Cluster 10:	Cluster 11:	Cluster 12:	Cluster 13:
HCC_DocW_LegallyBlonde4	VCT_Doc9_DirtyGrandpa	VCT_Doc6_DirtyGrandpa	ACB_Doc2_Grown Ups	CFA_Doc3_DespicableMe3	MDP_Doc6_Taxi	HCC_DocW_LegallyBlonde1
GCS_Doc3_LostCity	HCC_DocW_LegallyBlonde5	MDP_Doc10_Taxi	ACB_Doc7_Grown Ups	ACB_Doc4_Grown Ups	KCM_Doc5_ItChapterTwo	WCU_Doc3_NoTimeToDie
WCU_Doc9_NoTimeToDie	HCC_DocW_LegallyBlonde6	JGS_Doc4_Taken	VCT_Doc8_DirtyGrandpa	WCU_Doc4_NoTimeToDie		KCM_Doc1_ItChapterTwo
JGS_Doc2_Taken	GCS_Doc4_LostCity	YWM_Doc7_Batman	GCS_Doc6_LostCity	WCU_Doc5_NoTimeToDie		KCM_Doc10_ItChapterTwo
AJN_Doc9_TheToxicAvenger	JGS_Doc1_Taken		GCS_Doc10_LostCity	WCU_Doc10_NoTimeToDie		TWH_Doc6_PitchBlack
XSP_Doc3_MinorityReport	KCM_Doc2_ItChapterTwo		DWH_Doc10_Inception	MDP_Doc3_Taxi		
RTD_Doc4_Oblivion	KCM_Doc9_ItChapterTwo		KCM_Doc3_ItChapterTwo	DWH_Doc6_Inception		
	NPO_Doc2_DragMeToHell		KCM_Doc8_ItChapterTwo	DWH_Doc7_Inception		
	XSP_Doc7_MinorityReport		DSP_Doc6_Us	NPO_Doc3_DragMeToHell		
	RTD_Doc8_Oblivion		NPO_Doc9_DragMeToHell	NPO_Doc7_DragMeToHell		
	YWM_Doc4_Batman		NPO_Doc10_DragMeToHell	KST_Doc1_EEAaO		
	TWH_Doc4_PitchBlack		KST_Doc7_EEAaO	RTD_Doc10_Oblivion		

Cluster 14:	Cluster 15:	Cluster 16:	Cluster 17:	Cluster 18:	Cluster 19:
ACB_Doc3_Grown Ups	VCT_Doc4_DirtyGrandpa	CFA_Doc10_DespicableMe3	GCS_Doc1_LostCity	CFA_Doc4_DespicableMe3	CFA_Doc5_DespicableMe3
ACB_Doc6_Grown Ups	HCC_DocW_LegallyBlonde3	DSP_Doc2_Us	DWH_Doc9_Inception	ACB_Doc8_Grown Ups	CFA_Doc9_DespicableMe3
ACB_Doc10_Grown Ups	WCU_Doc6_NoTimeToDie	DSP_Doc8_Us	LLQ_Doc5_AngelHasFallen	VCT_Doc2_DirtyGrandpa	HCC_DocW_LegallyBlonde9
VCT_Doc1_DirtyGrandpa	JGS_Doc3_Taken	NPO_Doc5_DragMeToHell	TRH_Doc5_Fresh	VCT_Doc3_DirtyGrandpa	MDP_Doc1_Taxi
GCS_Doc7_LostCity	KCM_Doc7_ItChapterTwo	TRH_Doc1_Fresh	XSP_Doc4_MinorityReport	VCT_Doc10_DirtyGrandpa	MDP_Doc4_Taxi
GCS_Doc9_LostCity	YWM_Doc2_Batman	RTD_Doc2_Oblivion	TWH_Doc9_PitchBlack	HCC_DocW_LegallyBlonde2	JGS_Doc5_Taken
WCU_Doc2_NoTimeToDie				GCS_Doc8_LostCity	LLQ_Doc4_AngelHasFallen
MDP_Doc5_Taxi				JGS_Doc9_Taken	AJN_Doc5_TheToxicAvenger
DWH_Doc5_Inception				DWH_Doc4_Inception	DSP_Doc7_Us
LLQ_Doc9_AngelHasFallen				DWH_Doc8_Inception	DSP_Doc9_Us
NPO_Doc8_DragMeToHell				LLQ_Doc1_AngelHasFallen	TRH_Doc4_Fresh
XSP_Doc2_MinorityReport				LLQ_Doc3_AngelHasFallen	RTD_Doc7_Oblivion
RTD_Doc5_Oblivion				LLQ_Doc7_AngelHasFallen	YWM_Doc6_Batman
YWM_Doc8_Batman				KCM_Doc4_ItChapterTwo	YWM_Doc10_Batman
				AJN_Doc1_TheToxicAvenger	TWH_Doc5_PitchBlack
				TRH_Doc9_Fresh	
				KST_Doc10_EEAaO	
				KST_Doc3_EEAaO	
				XSP_Doc8_MinorityReport	
				RTD_Doc6_Oblivion	
				TWH_Doc8_PitchBlack	

Another method for grouping similar documents into clusters is topic modeling. LSA employs the Single Value Decomposition approach, whereas LDA use probabilistic methods to determine document similarity.

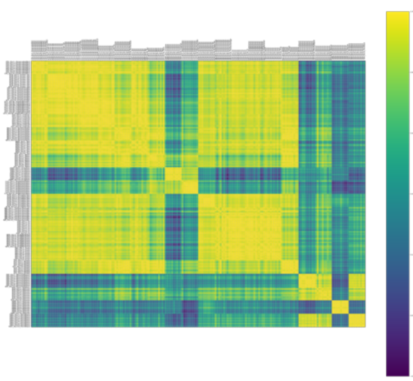
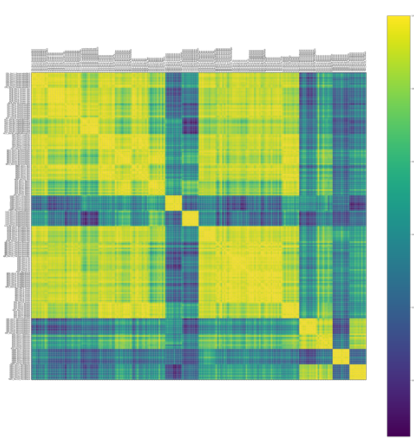
LSA model coherence results on the class corpus:

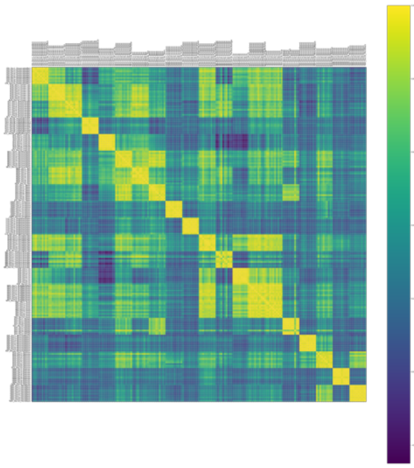
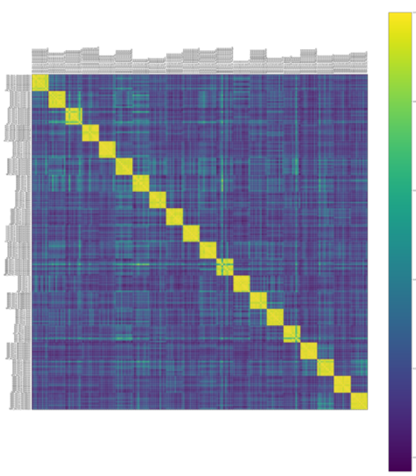
4 topics 10 words: 0.39437203127519627

5 topics 10 words: 0.35590603717536956

10 topics 10 words: 0.4221141433463105

20 topics 10 words: 0.4000300587151389

LSA (Latent Semantic Analysis)	
Topics 4 Words 10 	[(0, '0.240*"character" + 0.165*"first" + 0.140*"action" + 0.136*"scene" + 0.135*"would" + 0.128*"thing" + 0.127*"movie" + 0.122*"year" + 0.107*"doesnt" + 0.107*"batman"'), (1, '0.604*"batman" + 0.239*"penguin" + 0.229*"burton" + 0.176*"return" + 0.168*"catwoman" + 0.161*"gotham" + 0.145*"shreck" + 0.145*"dream" + 0.108*"christopher" + 0.104*"villain"'), (2, '0.241*"dream" + 0.195*"action" + 0.194*"inception" + -0.160*"batman" + 0.145*"anderton" + 0.138*"future" + -0.134*"funny" + -0.134*"comedy" + 0.132*"cruise" + 0.127*"spielberg"'), (3, '-0.304*"dream" + -0.230*"inception" + -0.186*"fallen" + -0.183*"banning" + 0.182*"black" + -0.182*"action" + 0.165*"pitch" + 0.157*"planet" + 0.155*"alien" + 0.136*"riddick"')]
Topic 5 Words 10 	[(0, '0.240*"character" + 0.165*"first" + 0.140*"action" + 0.136*"scene" + 0.135*"would" + 0.128*"thing" + 0.127*"movie" + 0.122*"year" + 0.107*"doesnt" + 0.107*"batman"'), (1, '0.604*"batman" + 0.239*"penguin" + 0.229*"burton" + 0.176*"return" + 0.168*"catwoman" + 0.161*"gotham" + 0.145*"shreck" + 0.145*"dream" + 0.108*"christopher" + 0.103*"villain"'), (2, '0.240*"dream" + 0.195*"action" + 0.194*"inception" + -0.161*"batman" + 0.144*"anderton" + 0.139*"future" + -0.134*"comedy" + -0.133*"funny" + 0.132*"cruise" + 0.127*"spielberg"'), (3, '-0.305*"dream" + -0.230*"inception" + -0.186*"fallen" + -0.184*"banning" + 0.182*"black" + -0.181*"action" + 0.165*"pitch" + 0.156*"alien" + 0.155*"planet" + 0.136*"riddick"'), (4, '-0.366*"fallen" + -0.341*"banning" + 0.285*"dream" + -0.222*"angel" + 0.212*"inception" + -0.193*"president" + 0.126*"nolan" + -0.123*"butler" + 0.105*"blonde" + -0.104*"agent"')]

Topic 10 Words 10 	[(0, '-0.240*"character" + -0.165*"first" + -0.140*"action" + -0.136*"scene" + -0.135*"would" + -0.128*"thing" + -0.127*"movie" + -0.122*"year" + -0.107*"doesnt" + -0.107*"batman"'), (1, '-0.604*"batman" + -0.239*"penguin" + -0.229*"burton" + -0.176*"return" + -0.169*"catwoman" + -0.161*"gotham" + -0.145*"shreck" + -0.145*"dream" + -0.108*"christopher" + -0.103*"villain"'), (2, '-0.241*"dream" + -0.195*"action" + -0.194*"inception" + 0.161*"batman" + -0.144*"anderton" + -0.139*"future" + 0.135*"comedy" + 0.134*"funny" + -0.132*"cruise" + -0.127*"spielberg"'), (3, '-0.304*"dream" + -0.230*"inception" + -0.187*"fallen" + -0.183*"banning" + 0.182*"black" + -0.181*"action" + 0.165*"pitch" + 0.156*"planet" + 0.156*"alien" + 0.136*"riddick"'), (4, '-0.367*"fallen" + -0.340*"banning" + 0.285*"dream" + -0.222*"angel" + 0.211*"inception" + -0.192*"president" + 0.126*"nolan" + -0.122*"butler" + 0.105*"blonde" + -0.104*"agent"'), (5, '0.209*"blonde" + 0.182*"witherspoon" + -0.177*"evelyn" + 0.168*"school" + 0.154*"warner" + 0.153*"legally" + -0.140*"family" + 0.138*"banning" + 0.134*"murder" + 0.133*"harvard"'), (6, '0.199*"anderton" + -0.180*"black" + -0.172*"dream" + 0.166*"spielberg" + -0.165*"pitch" + 0.165*"report" + 0.164*"future" + 0.160*"precrime" + 0.159*"evelyn" + 0.151*"minority"'), (7, '-0.228*"horror" + 0.225*"loretta" + 0.199*"bullock" + -0.192*"family" + -0.185*"peele" + 0.169*"tatum" + 0.150*"daniel" + 0.130*"doesnt" + 0.122*"evelyn" + 0.121*"character"'), (8, '0.206*"evelyn" + 0.200*"blonde" + 0.168*"witherspoon" + -0.166*"loretta" + 0.158*"everywhere" + 0.153*"everything" + -0.153*"bullock" + 0.152*"warner" + 0.145*"legally" + 0.140*"school"'), (9, '-0.382*"toxic" + -0.261*"avenger" + -0.188*"melvin" + 0.175*"loretta" + 0.160*"bullock" + 0.155*"despicable" + 0.134*"tatum" + -0.131*"waste" + 0.120*"minion" + 0.112*"family"')]
Topics 20 Words 10 	[(0, '-0.240*"character" + -0.165*"first" + -0.140*"action" + -0.136*"scene" + -0.135*"would" + -0.128*"thing" + -0.127*"movie" + -0.122*"year" + -0.107*"doesnt" + -0.107*"batman"'), (1, '0.604*"batman" + 0.239*"penguin" + 0.229*"burton" + 0.176*"return" + 0.168*"catwoman" + 0.161*"gotham" + 0.145*"shreck" + 0.145*"dream" + 0.108*"christopher" + 0.103*"villain"'), (2, '-0.240*"dream" + -0.195*"action" + -0.194*"inception" + 0.160*"batman" + -0.145*"anderton" + -0.138*"future" + 0.135*"comedy" + 0.134*"funny" + -0.132*"cruise" + -0.127*"spielberg"'), (3, '0.305*"dream" + 0.230*"inception" + 0.186*"fallen" + 0.183*"banning" + 0.182*"action" + -0.182*"black" + -0.164*"pitch" + -0.156*"planet" + -0.156*"alien" + -0.136*"riddick"'), (4, '0.366*"fallen" + 0.340*"banning" + -0.286*"dream" + 0.222*"angel" + -0.211*"inception" + 0.192*"president" + -0.127*"nolan" + 0.122*"butler" + -0.105*"blonde" + 0.104*"agent"'), (5, '0.208*"blonde" + 0.182*"witherspoon" + -0.177*"evelyn" + 0.168*"school" + 0.154*"warner" + 0.153*"legally" + -0.140*"family" + 0.138*"banning" + 0.134*"murder" + 0.133*"harvard"'), (6, '0.199*"anderton" + -0.179*"black" + -0.171*"dream" + 0.165*"spielberg" + -0.165*"pitch" + 0.165*"report" + 0.162*"future" + 0.160*"evelyn" + 0.159*"precrime" + 0.151*"minority"'),

	(7, '0.230*"horror" + -0.225*"loretta" + -0.200*"bullock" + 0.191*"family" + 0.187*"peele" + -0.169*"tatum" + -0.148*"daniel" + -0.130*"doesnt" + -0.121*"evelyn" + -0.121*"character"'), (8, '-0.207*"evelyn" + -0.200*"blonde" + -0.169*"witherspoon" + 0.165*"loretta" + -0.157*"everywhere" + -0.153*"everything" + 0.152*"bullock" + -0.152*"warner" + -0.145*"legally" + -0.141*"school"'), (9, '0.383*"toxic" + 0.262*"avenger" + 0.189*"melvin" + -0.173*"loretta" + -0.159*"bullock" + -0.156*"despicable" + -0.133*"tatum" + 0.132*"waste" + -0.121*"minion" + -0.113*"family"'), (10, '-0.219*"horror" + 0.215*"grandpa" + -0.201*"toxic" + 0.199*"jason" + 0.189*"despicable" + -0.186*"loretta" + -0.163*"bullock" + 0.155*"dirty" + -0.142*"avenger" + 0.138*"minion"'), (11, '-0.332*"grandpa" + -0.327*"jason" + 0.250*"despicable" + -0.230*"dirty" + 0.190*"minion" + 0.133*"bratt" + 0.123*"toxic" + -0.109*"efron" + -0.103*"horror" + -0.102*"peele"'), (12, '0.229*"toxic" + -0.207*"james" + 0.196*"despicable" + 0.153*"minion" + -0.151*"chapter" + 0.149*"avenger" + -0.146*"loser" + 0.134*"family" + -0.125*"scene" + -0.123*"year"'), (13, '0.227*"christine" + 0.203*"raimi" + 0.198*"bryan" + -0.195*"family" + 0.176*"taken" + -0.158*"sandler" + 0.150*"neeson" + 0.145*"woman" + -0.134*"peele" + -0.132*"james"'), (14, '-0.288*"bryan" + -0.286*"taken" + -0.218*"neeson" + -0.157*"peele" + 0.150*"christine" + 0.149*"evelyn" + 0.141*"woman" + 0.135*"raimi" + -0.133*"daughter" + -0.124*"paris"'), (15, '0.194*"grandpa" + 0.174*"loser" + -0.171*"fallon" + 0.164*"chapter" + -0.161*"peele" + -0.153*"funny" + 0.152*"jason" + -0.150*"action" + -0.149*"queen" + -0.145*"latifah"'), (16, '0.284*"christine" + 0.255*"raimi" + -0.229*"fresh" + -0.227*"steve" + -0.156*"dating" + -0.143*"first" + 0.141*"would" + 0.125*"raimis" + 0.124*"gypsy" + 0.114*"david"'), (17, '-0.237*"sandler" + -0.181*"fresh" + -0.173*"woman" + -0.169*"steve" + 0.167*"fallon" + -0.152*"grown" + 0.144*"latifah" + 0.139*"queen" + 0.135*"jimmy" + -0.131*"spade"'), (18, '0.418*"oblivion" + 0.224*"cruise" + 0.203*"earth" + 0.139*"drone" + 0.138*"harper" + 0.135*"kosinski" + -0.126*"black" + 0.124*"movie" + -0.110*"fresh" + 0.104*"joseph"'), (19, '0.239*"craig" + 0.230*"craigs" + 0.187*"daniel" + 0.161*"james" + 0.157*"malek" + 0.151*"madeleine" + 0.143*"bond" + 0.137*"thing" + 0.127*"franchise" + 0.122*"safin"')]

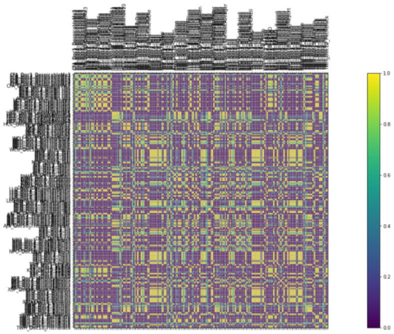
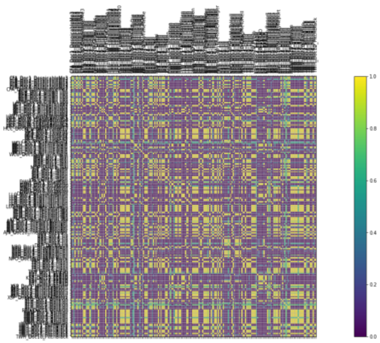
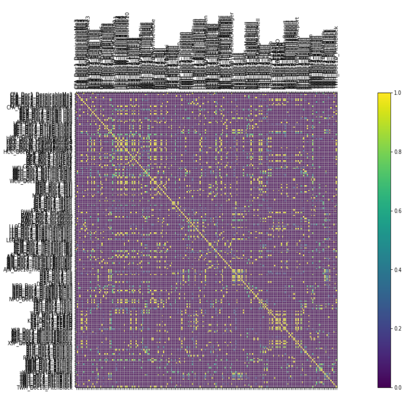
LDA model coherence results on the class corpus:

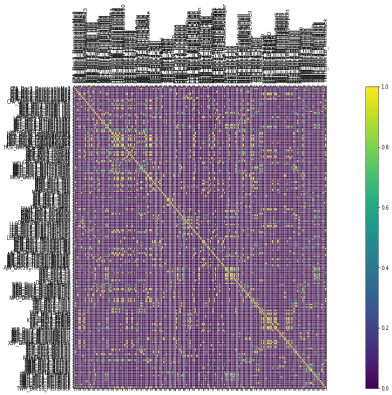
4 topics 10 words: 0.24335388081086468

5 topics 10 words: 0.23510885809373305

10 topics 10 words: 0.24692908290514537

20 topics 10 words: 0.2650688095351848

LDA	
Topic 4 Words 10 	[(0, '0.003*"would" + 0.003*"character" + 0.003*"movie" + 0.002*"scene" + 0.002*"doesnt" + 0.002*"first" + 0.002*"batman" + 0.002*"year" + 0.002*"comedy" + 0.002*"never"'), (1, '0.005*"character" + 0.003*"family" + 0.003*"year" + 0.003*"action" + 0.003*"first" + 0.002*"jason" + 0.002*"thing" + 0.002*"point" + 0.002*"grandpa" + 0.002*"fallen"'), (2, '0.004*"character" + 0.003*"first" + 0.003*"thing" + 0.003*"horror" + 0.003*"action" + 0.002*"dream" + 0.002*"movie" + 0.002*"there" + 0.002*"batman" + 0.002*"year"'), (3, '0.005*"character" + 0.004*"first" + 0.003*"scene" + 0.003*"action" + 0.002*"funny" + 0.002*"thats" + 0.002*"horror" + 0.002*"thing" + 0.002*"every" + 0.002*"there"')]
Topic 5 Words 10 	[(0, '0.004*"character" + 0.003*"would" + 0.002*"dream" + 0.002*"thing" + 0.002*"black" + 0.002*"year" + 0.002*"family" + 0.002*"comedy" + 0.002*"action" + 0.002*"james"'), (1, '0.005*"character" + 0.003*"first" + 0.003*"funny" + 0.003*"scene" + 0.003*"there" + 0.003*"woman" + 0.002*"little" + 0.002*"thing" + 0.002*"would" + 0.002*"another"'), (2, '0.003*"thing" + 0.003*"first" + 0.003*"character" + 0.003*"scene" + 0.003*"would" + 0.002*"family" + 0.002*"year" + 0.002*"movie" + 0.002*"action" + 0.002*"really"'), (3, '0.004*"batman" + 0.004*"character" + 0.003*"first" + 0.003*"movie" + 0.003*"action" + 0.003*"horror" + 0.003*"still" + 0.003*"thing" + 0.002*"would" + 0.002*"scene"'), (4, '0.003*"character" + 0.003*"fallen" + 0.003*"dream" + 0.003*"action" + 0.003*"evelyn" + 0.003*"scene" + 0.002*"movie" + 0.002*"everything" + 0.002*"world" + 0.002*"banning"')]
Topic 10 Words 10 	[(0, '0.007*"batman" + 0.004*"character" + 0.003*"penguin" + 0.003*"would" + 0.003*"going" + 0.003*"action" + 0.003*"first" + 0.003*"dream" + 0.003*"year" + 0.003*"movie"'), (1, '0.006*"character" + 0.004*"thing" + 0.003*"banning" + 0.003*"scene" + 0.003*"still" + 0.003*"action" + 0.003*"funny" + 0.003*"there" + 0.003*"future" + 0.002*"bullock"'), (2, '0.005*"would" + 0.005*"character" + 0.005*"family" + 0.003*"despicable" + 0.003*"scene" + 0.003*"year" + 0.003*"first" + 0.003*"world" + 0.003*"action" + 0.003*"funny"'), (3, '0.005*"batman" + 0.003*"character" + 0.003*"year" + 0.003*"thing" + 0.002*"together" + 0.002*"scene" + 0.002*"family" + 0.002*"movie" + 0.002*"going" + 0.002*"director"'), (4, '0.005*"character" + 0.005*"horror" + 0.004*"toxic" + 0.004*"comedy" + 0.003*"blonde" + 0.003*"woman" + 0.003*"first" + 0.003*"point" + 0.002*"movie" + 0.002*"thing"'), (5, '0.004*"first" + 0.003*"character" + 0.003*"daniel" + 0.003*"doesnt" + 0.003*"everything" + 0.003*"everywhere" + 0.003*"family" + 0.003*"toxic" + 0.002*"space" + 0.002*"planet"'), (6, '0.005*"scene" + 0.005*"christine" + 0.004*"raimi" + 0.003*"horror" + 0.003*"woman" + 0.003*"first" + 0.003*"something" + 0.002*"movie" + 0.002*"action" + 0.002*"moment"'), (7, '0.004*"character" + 0.004*"action" + 0.004*"first" + 0.003*"thing" + 0.003*"would" + 0.003*"daughter" + 0.003*"james" + 0.003*"jason" + 0.003*"taken" + 0.002*"doesnt"'),

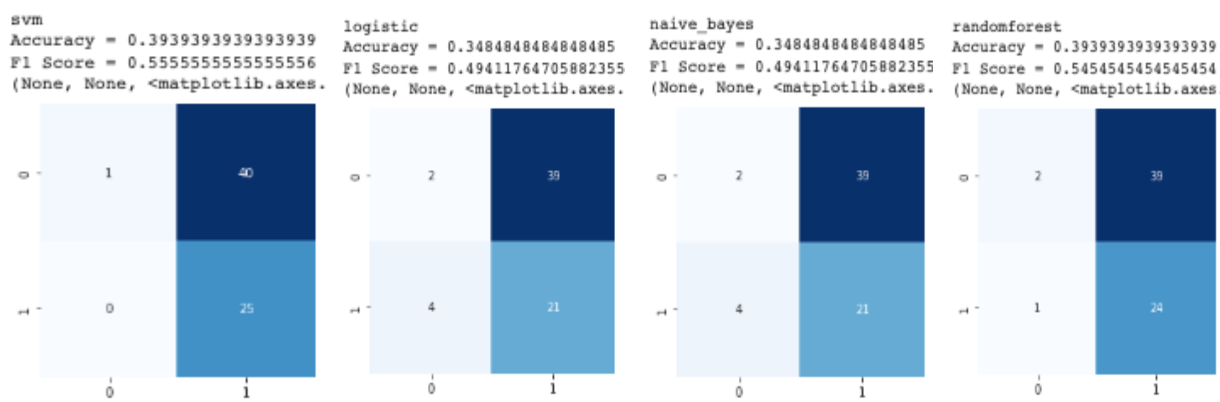
	(8, '0.006*"first" + 0.005*"dream" + 0.003*"character" + 0.003*"inception" + 0.003*"movie" + 0.002*"going" + 0.002*"horror" + 0.002*"people" + 0.002*"woman" + 0.002*"black"'), (9, '0.003*"character" + 0.003*"black" + 0.003*"pitch" + 0.003*"alien" + 0.003*"riddick" + 0.003*"planet" + 0.003*"woman" + 0.002*"year" + 0.002*"movie" + 0.002*"david"')]
Topic 20 Words 10 	[(0, '0.006*"character" + 0.004*"dream" + 0.003*"movie" + 0.003*"scene" + 0.003*"first" + 0.003*"doesnt" + 0.003*"could" + 0.003*"oblivion" + 0.003*"steve" + 0.002*"woman"'), (1, '0.006*"character" + 0.005*"first" + 0.004*"family" + 0.004*"horror" + 0.003*"never" + 0.003*"dream" + 0.003*"bryan" + 0.003*"daughter" + 0.003*"three" + 0.003*"batman"'), (2, '0.004*"funny" + 0.004*"first" + 0.003*"thing" + 0.003*"character" + 0.003*"murder" + 0.003*"point" + 0.003*"great" + 0.003*"anderton" + 0.003*"would" + 0.003*"every"'), (3, '0.009*"bryan" + 0.006*"paris" + 0.005*"neeson" + 0.005*"daughter" + 0.004*"taken" + 0.004*"world" + 0.003*"besson" + 0.003*"friend" + 0.003*"action" + 0.003*"parent"'), (4, '0.006*"batman" + 0.004*"action" + 0.004*"taken" + 0.004*"character" + 0.003*"woman" + 0.003*"thing" + 0.003*"would" + 0.003*"christine" + 0.003*"first" + 0.002*"movie"'), (5, '0.006*"character" + 0.005*"evelyn" + 0.004*"scene" + 0.004*"everything" + 0.003*"waymond" + 0.003*"movie" + 0.003*"loretta" + 0.003*"daniel" + 0.003*"everywhere" + 0.003*"woman"'), (6, '0.005*"thing" + 0.004*"still" + 0.003*"raimi" + 0.003*"action" + 0.003*"grandpa" + 0.003*"character" + 0.003*"woman" + 0.002*"fallen" + 0.002*"another" + 0.002*"horror"'), (7, '0.006*"peele" + 0.004*"comedy" + 0.004*"family" + 0.003*"spielberg" + 0.003*"character" + 0.003*"future" + 0.003*"double" + 0.003*"nyong" + 0.003*"year" + 0.003*"funny"'), (8, '0.004*"action" + 0.003*"first" + 0.003*"cruise" + 0.003*"effect" + 0.003*"thing" + 0.003*"woman" + 0.003*"batman" + 0.002*"futuristic" + 0.002*"still" + 0.002*"oblivion"'), (9, '0.004*"year" + 0.004*"first" + 0.004*"character" + 0.003*"scene" + 0.003*"batman" + 0.003*"around" + 0.002*"grown" + 0.002*"james" + 0.002*"another" + 0.002*"there"'), (10, '0.006*"banning" + 0.005*"president" + 0.004*"action" + 0.004*"character" + 0.004*"doesnt" + 0.004*"everything" + 0.003*"first" + 0.003*"thing" + 0.003*"daniel" + 0.003*"dream"'), (11, '0.005*"batman" + 0.004*"oblivion" + 0.004*"penguin" + 0.003*"first" + 0.003*"character" + 0.003*"taken" + 0.003*"action" + 0.003*"world" + 0.003*"burton" + 0.003*"human"'), (12, '0.006*"friend" + 0.006*"family" + 0.005*"comedy" + 0.003*"great" + 0.003*"thing" + 0.003*"black" + 0.003*"together" + 0.003*"humor" + 0.003*"night" + 0.003*"thats"'), (13, '0.006*"would" + 0.004*"scene" + 0.004*"character" + 0.003*"doesnt" + 0.003*"year" + 0.003*"little" + 0.003*"movie" + 0.003*"loretta" + 0.003*"grandpa" + 0.003*"really"'), (14, '0.003*"character" + 0.003*"there" + 0.002*"dating" + 0.002*"fresh" + 0.002*"fantasy" + 0.002*"steve" + 0.002*"scavs" + 0.002*"audience" + 0.002*"jason" + 0.002*"doesnt"'), (15, '0.004*"family" + 0.004*"fallen" + 0.004*"dream" + 0.004*"action" + 0.004*"black" + 0.004*"adelaide" + 0.003*"peele" + 0.003*"character" + 0.003*"pitch" + 0.003*"inception"'), (16, '0.004*"first" + 0.004*"character" + 0.004*"taste" + 0.003*"going" + 0.003*"action" + 0.003*"movie" + 0.003*"there" + 0.003*"something" + 0.003*"never" + 0.003*"dream"'),

	(17, '0.006*"character" + 0.004*"funny" + 0.004*"first" + 0.004*"comedy" + 0.004*"would" + 0.004*"woman" + 0.004*"year" + 0.003*"scene" + 0.003*"horror" + 0.003*"really"'), (18, '0.005*"loser" + 0.004*"first" + 0.003*"really" + 0.003*"world" + 0.002*"scene" + 0.002*"thats" + 0.002*"character" + 0.002*"original" + 0.002*"evelyns" + 0.002*"group"'), (19, '0.004*"scene" + 0.004*"point" + 0.003*"avenger" + 0.003*"chapter" + 0.003*"might" + 0.003*"first" + 0.003*"thats" + 0.003*"take" + 0.003*"loretta" + 0.002*"horror"')]

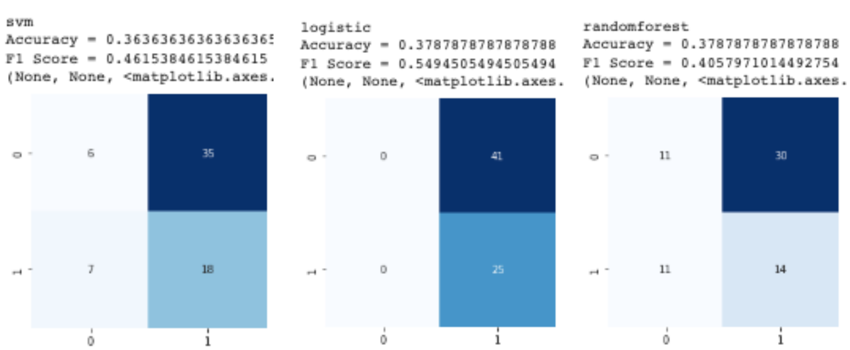
We used the entire class corpus for sentiment analysis method and classified positive and negative reviews using the TF IDF and doc2vec vectorization approaches. The dataset was splits into 67% train dataset and 33% test dataset through these model was performed to predict movie reviews. The following discoveries were made:

	Support Vector Machine		Logistic Regression		Naïve Bayes		Random Forest	
	Accuracy	**F1 Score**	**Accuracy**	**F1 Score**	**Accuracy**	**F1 Score**	**Accuracy**	**F1 Score**
TF-IDF	0.393939394	0.555555556	0.348484848	0.494117647	0.348484848	0.49411765	0.393939394	0.545454545
Doc2Vec 300	0.363636364	0.461538462	0.378787879	0.549450549			0.378787879	0.405797101
Doc2Vec 600	0.363636364	0.461538462	0.378787879	0.549450549			0.393939394	0.428571429

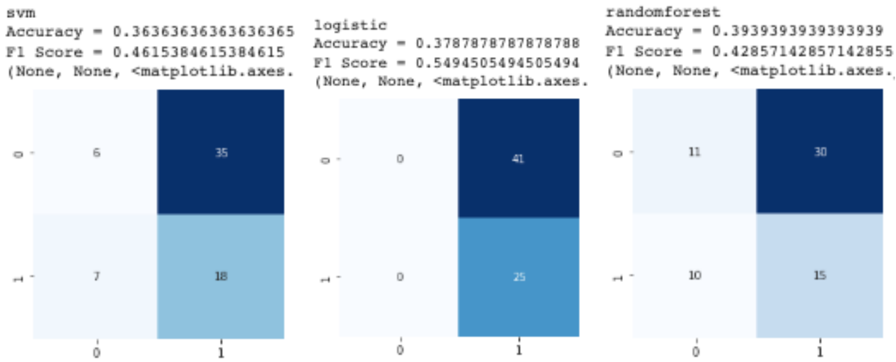
Confusion matrix Based on TF-IDF: svm, logistic, Naïve Bayes random forest



Confusion matrix Based on doc2vec vector size 300: svm, logistic, and random forest



Confusion matrix Based on doc2vec vector size 600: svm, logistic, and random forest



Analysis and Interpretation

LSA for four topics produced no distinct patterns that would distinguish the four movie genres. The enormous square on the upper left features a few movies from all genres, while the bottom right square has a mix of several movie genres as well. However, as demonstrated by the prominent diagonal line in the heatmap for 20 topics, LSA revealed the difference between each individual movie based on the titles. The top words for each topic were strongly correlated to the individual movie assigned to that topic. LDA showed no evident pattern the results depict the outcomes of 2, 4, and 20 documents utilizing 10 words and 200 epochs, respectively. Due to the smaller size of the corpus class, LDA did not perform as well as LSA. More data for each topic may help the LDA model.

One of this analysis goals was to group similar documents into clusters. Overall, regardless of what k was, the silhouette scores never averaged above 0.09. Clustering plots were created to display the clusters. K-means clustering method produced by the tf-idf performed well on k-cluster size 2. The first cluster contained mostly action and sci-fi movies categories. In contrast, the second cluster contained mostly comedy and horror genres with only few action, and sci-fi movie documents. So, we could say that we have two defined clusters, one with horror and comedy and the other with action and sci-fi. This cluster made sense as some of the movies in the comedy category are “dark” comedies. I also thought that it was interesting that although my movie (Us) was correctly classified in the Horror genre with It Chapter Two, Fresh, and DragMetoHell. It was also clustered with the Comedy movie DespicableMe3 as well although this isn't accurate clustering results, I thought it made sense because Us has several comedy bits throughout the movies as it includes a family part correlation in both movies. The findings for K-means clustering method produced by the tf-idf performed well on k-cluster size 4 indicated that cluster 0 and 3 are not segmented properly since it contains a mix of all genres. Similarly, cluster 1 contains both Action and Sci-Fi movies genres. Based on our observation only cluster 2 was segmented properly it has only Sci-Fi genre movie reviews as expected. The clustering on 20 clusters performed very well, grouping the reviews on 20 categories based on their titles. In addition, it evaluated properly based on Silhouette scores. This allowed me to determine that k=20 was the ideal cluster size since performance "peaked" at these numbers. Since there are 20 movie titles in all, it makes it logical to proceed with k=20.

Classification algorithms were examined to determine how effectively they could identify sentiment: positive and negative movie reviews. These experiments used the Naïve Bayes, SVM, and Random Forest models based on TF-IDF and Doc2Vec with embedding vector dimensions of 300 and 600. Changing the number of dimensions had no evident effect on the results. All models had their hyperparameters tuned, and the total accuracy and F1-score were recorded for comparison. Random forest slightly outperformed the other classifiers in terms of accuracy and F1 score, with values greater than 40% for all TF-IDF, doc2vec 300, and doc2vec 600. Reading the confusion matrices for TF-IDF, we discovered that 2 reviews were accurately labeled as negative, and 24 reviews were correctly classified as positive in the case of random forest. Similarly, 11 reviews were accurately identified as negative and 14 as positive for doc2vec 300 dimensions. For doc2vec 600 dimensions, 11 reviews were accurately identified as negative, while 15 were correctly classed as negative. Even while the random forest scored below average, it outperformed other classifiers, with total accuracy ratings that were just less than 50% of the likelihood of flipping a coin. More reviews may be required for the models to recognize the difference between negative and positive, or the reviews themselves may be insufficiently positive or negative. More reviews or greater positive/negative discrimination might improve the models' overall accuracy.

Conclusion

The research we conducted until this point suggests that a classification based on the movies' titles using K-Means Clustering unsupervised modeling would be our best choice. Our model performed well on 20 clusters on TF_IDF. Similarly, LSA method did provide some interesting insights that would be helpful based on the coherence results however, I would not recommend doc2vec method based on analysis using LDA method on the current dataset. Further, classification experiments regarding supervised model for predicted if the movie review based positive or negative labels requires more data, and further modification of the model to produce better accuracy. Also, I would recommend doing better splits of training and test dataset which I didn't do decent job in this assignment based on accuracy results it might have needed more training data then I had expected. However, 300 and 600 dimensions were tested and neither produced the expected clusters. From the classification point of view, random forest would be the best choice so far in terms of supervised modeling. Even though the values were a that were just less than 50% of the likelihood of flipping a coin, random forest performed better than the other classifiers.

References

Jurafsky, D. and Martin, J., 2019, Speech and Language Processing (3rd ed.draft)

Lane, H., C. Howard, and H. M. Hapke 2019. Natural Language Processing in Action: Understanding, Analyzing, and Generating Text with Python.

Python Wife. 2022. TF-IDF vectors in Natural Language Processing. Retrieved from https://pythonwife.com/tf-idf-vectors-in-natural-language-processing/