Project 5 – Advanced Data Mining Applications
CS548 / BCB503 Knowledge Discovery and Data Mining - Fall 2017
Prof. Carolina Ruiz

Student: Janvi Kothari

Description of the particular problem within the selected data mining topic to be addressed in this project	/15
Description of the approach used in this project to tackle the above problem. *All data mining techniques you use in this project for pre-processing, mining and evaluation must have been covered in class during this semester.*	/25
Description of the dataset selected	/15
Appropriateness of the dataset selected with respect to this topic/problem	/10
Guiding questions	/10
Preprocessing	/10
Experiments: Sufficient & coherent	/25
Objectives, Data, Additional Pre/Post-processing	/20
Presentation of results	/20
Analysis of results	/30
Overall discussion, comparisons, and conclusions	/20
TOTAL	/200

Total Written Report: ______________/200 = ___________/100
Class Presentation: ___________/100
Class participation during project presentation: ___________/100

Do not exceed the given page limits for this written report

Topic: Text Mining <at most 1 page>

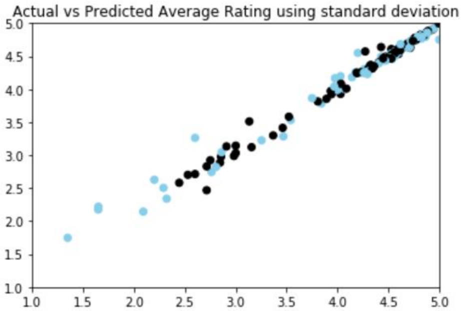
1. **Description of the particular problem within the selected data mining topic to be addressed in this project:** The problem for the topic of Text Mining is most importantly identify the significant terms present in the corpus, analyze & interpret them as to what meaning they give us about the text, predict about those significant terms and compare with already existing analysis, understand the sentiments behind the text. Overall, we want to understand if Mr. Trump's words influence the average rating of a video, how the sentiments have transitioned and how the speeches can be grouped over time.
2. **Description of the approach used in this project to tackle the above problem:** Firstly, for cleaning of the data used stopword removal, tokenization, lemmatization, feature extraction & to divide the data into different time frames, implemented binning. To find the results implemented, method used are: linear regression, Random Forests, scatterplot, sentiment analysis, clustering.
3. **Dataset Name:** Mr. Donald Trump's Speeches 4. **Where found**: https://www.kaggle.com/
4. **Dataset Description:** This data set consists of speeches of President Donald Trump that are available on Youtube from 01st Feb 2016 to 03rd August 2017(downloaded using youtube-dl package). This means data about his speeches is available from during the election campaign and during his presidency phase. This dataset consists of the speeches as subtitles extracted from the videos i.e. it is only a stream of words being translated from speech into text by GoogleVoice. The corpus contains 836 total documents having 9 attributes such as:
   ID: Unique ID of the video; **Upload_date**: The date when the video was uploaded on youtube.; **Title**: Title of the video; **View_count**: Number of views on the video/speech; **Like_count**: Total number of likes on the video/speech; **Dislike_count**: Total number of dislikes on the video/speech; **Playlist**: Playlist to which the videos belong ("Donald Trump Speeches & Events", "DONALD TRUMP SPEECHES & PRESS CONFERENCE", "President Donald Trump Weekly Address 2017", "President Donald Trump's First 100 Days | NBC News", "Donald Trump Rally Speech Events Press Conference Rallies Playlist"); **Subtitles**: The translated text from the video; **Average_Rating**: Rating of each video.(range from 1.0 to 5.0 float values)
   The corpus consists of with each document consisting range of 37 to 83482 words.
5. **Initial data preprocessing, if any:** For initial pre-processing, eliminated any punctuations or symbols other than the English alphabets and converted all to lowercase text(using re), removed stop words(using nltk.stopwords), tokenization of words(using word_tokenize from nltk.tokenize), stemming of words(using WordNetLemmatizer from nltk.stem), feature extraction(using TfidfVectorizer from sklearn.feature_extraction.text). Besides this, for analysis have used the upload_date attribute to help create time based ranges between the data. (Will be explained further for each experiment).
6. **Three Guiding Questions about the dataset domain:**

1. Given all the speeches, how well can we predict the average rating for each speech based on important features from the speech and the count of likes and dislikes?
2. How well can we understand the change of sentiments of over a period in Mr. Trump's speeches? Is there any significant change before and after his presidency?
3. Are we able to find any groups in the data using the speeches based on the timeline of pre-election campaign phase, post-election phase and presidential phase?

	Summary of Experiments. *At most 2 page.*							
Q	**Tool**	**Pre-processing**	**Mining Technique**	**Parameters**	**Results**	**Time taken**	**Evaluation**	**Observations about experiment** **Observations about visualization** **Interpretation results**
1	Python	Get weights of extracted terms after feature extraction(tfidf)	**Text Mining & Regression** Linear Regression (sklearn.linear_model - LinearRegression)	Standard deviation of weights of terms per document, dislike_count, like_count (min_df=5,max_df=0.95,ngram_range=(2, 3),max_features=6000)	Mean ratings Predicted =4.34 Actual=4.33	0.2987 seconds	MSE = 0.4931	Once the important terms were extracted as features, the weights i.e. the idf values for every term are used as scores for each document. This is used along with like and dislike count for each document as other parameters for regression. This gives a good predicted overall average rating; however, the MSE is 0.493 which is not that good a result considering that a rating with 4.0 can be predicted to 3.5 or 4.5. This would change the dynamic of how a person would perceive a video.
1	Python	Get weights of extracted terms after feature extraction(tfidf)	**Text Mining & Regression** Random Forest (sklearn.ensemble - RandomForestRegressor)	Standard deviation of weights of terms per document, dislike_count, like_count	Mean ratings Predicted = 4.35 Actual=4.33	0.4772 seconds	MSE = 0.0116	Similarly, when we apply random forests to predict the average rating for the video, it does a good job considering the MSE is 0.0116. The scatter plot for the same shows us a positive correlation of actual versus predicted average rating, and there's almost a linear relationship here.
1	Python	Get weights of extracted terms after feature extraction(tfidf)	**Text Mining & Regression** Linear Regression (sklearn.linear_model)	Sum,mean,median,standard deviation of weights of terms per document, dislike_count, like_count	Mean ratings Predicted = 4.35 Actual=4.33	0.1734 seconds	MSE = 0.4437	Similarly, along with sum of weights, its mean, median, standard deviation are used along with like & dislike counts. Here MSE is 0.44 which also not that good a result. Overall, we can interpret that linear regression does not do a very efficient job compared to Random Forests in predicting the average rating.
1	Python	Get weights of extracted terms after feature extraction(tfidf)	**Text Mining & Regression** Random Forest (sklearn.ensemble)	Sum,mean,median,standard deviation weights of terms per document,dislikecount,likecount	Mean ratings Predicted = 4.35 Actual=4.33	0.4201 seconds	MSE = 0.0232	MSE when including all these parameters is slightly higher than when just implementing Random Forests using just standard deviation. Performance time of random forests is as expected higher than that of linear regression. The scatter plot for the same shows us a positive correlation of actual versus predicted average rating, but the plot is jittered.

2	Python	None	**Text Mining** Natural Language Processing (nltk.sentiment)		Average sentiments polarity=0.1604 p_pos=0.9822 p_neg=0.0177	12.2368 seconds 2hrs for entire		This experiment shows us the average sentiments that can be understood out of the total speeches. The speeches are overall neutral based on the polarity analysis (-1 to +1). Further looking at the p_pos and p_neg it can be interpreted that overall the speeches are highly positive and very slightly negative. (0 is smallest and 1 is highest)
2	Python	Divide data into two periods as Before Presidency and After Presidency using upload_date	**Text Mining** Natural Language Processing (nltk.sentiment)		Average Sentiments Before:After polarity= 0.1527: 0.1664 p_pos= 0.9933 : 0.9731 p_neg= 0.0066 : 0.0268	11.38 seconds		Here we can see based on the polarity that speeches are usually neutral both before and after election results. However, when we interpret the p_neg the value that we achieve after Mr. Trump being selected as the President are quite higher than that of before. Also, positivity has slightly reduced overall from Before to After election.
3	Python	Divide data into 5 periods using upload_date	**Text Mining & Clustering** K-means	K=5	The clusters we find are monotonous.	0.04511 seconds	Silhouette score = 0.1419	The labels that we achieve out of clustering are mapped to labels that we have created from upload_date such as Republican National Convention Campaigns, Republican Presidential Campaign, Elected as President, Inaugural Day & Presidential Tenure. Though the evaluation metric suggests that the clustering is poor, we observe that campaign speeches are in same clusters and presidential speeches are in same clusters.

Analysis of Results: (at most 1 page):



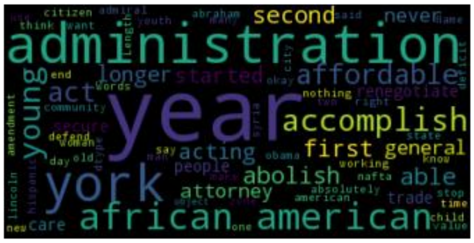
Scatter Plot of Predicted vs Actual Average Rating using SD of weights and count of likes & dislikes

For question 1, after the features are extracted, the weights of each term are used to create scores for each document. New parameters created are sum,mean,median,standard deviation of weights. On experimenting with only one of these parameters along with like & dislike counts for each video, we try to predict the average rating. It is observed that the best prediction is achieved using standard deviation of the weights, thus giving us the least MSE of 0.0116. However, when we compare the outcome from linear regression with random forests, though the time to perform random forests is slightly higher, results from it are much better. This also suggests that using just the term significance gives better results.

For question 2, as we can observe from the overall sentiments of the speeches Mr. Trump's words are highly positive and very rarely negative. Though we do identify that there is change in measure of negativity before and after his presidency.

For question 3, we try to find existing clusters in the data based on the sum of scores of features and the view counts & average rating for each video. We then compare with labels that we have created from upload_date using timeline information from https://www.aol.com/2016-election/timeline/. So the timeline that we create are:

Time period	Label
01st Feb 2016 to 20th July 2016	RNC-Campaign (Republican National Convention)
21st July to 8th Nov 2016	Presidential-Campaign (After accepting Presidential Nomination)
09th Nov 2016 to 19th Jan 2017	President-Elected (After election results were out)
20th Jan 2017	Inaugural-Day
21st Jan 2017 to 03rd August 2017	President



Word cloud shows important features obtained from the entire data set after removing stop words and vectorizing.

The observations from this experiment are that the speeches that are clustered together are campaign based and 'President' and 'Inaugural Day' are clustered together. This displays how there are changes in the speeches during different time durations of the election period. For further analysis just to understand more about text mining, I worked on trying to find important factors or topics from using these periods of time. We get amazing outcomes such as the important topics have modified from "China, Mexico" to "Obamacare, Tax deals" to "Women, Russia, Defense" to "America, People, Glorious" to "Administration, Job regulation". This is excellent analysis that have been obtained as we can see how there has been a transition in speeches by Mr. Trump over this phase.

Summary of what you learned in this project:

From this project I have been able to understand text mining in depth, how important features can be used quantitatively so as to use for prediction/classification/clustering. Also, I have understood about the working of sentiment analysis using nltk and natural language processing. Further I have gained knowledge about the election phase and how using the variation in time frame gives insight about the text and its transition over time.