Unsupervised Clustering of Election Documents

Rayan Jaipuriyar

August 23, 2024

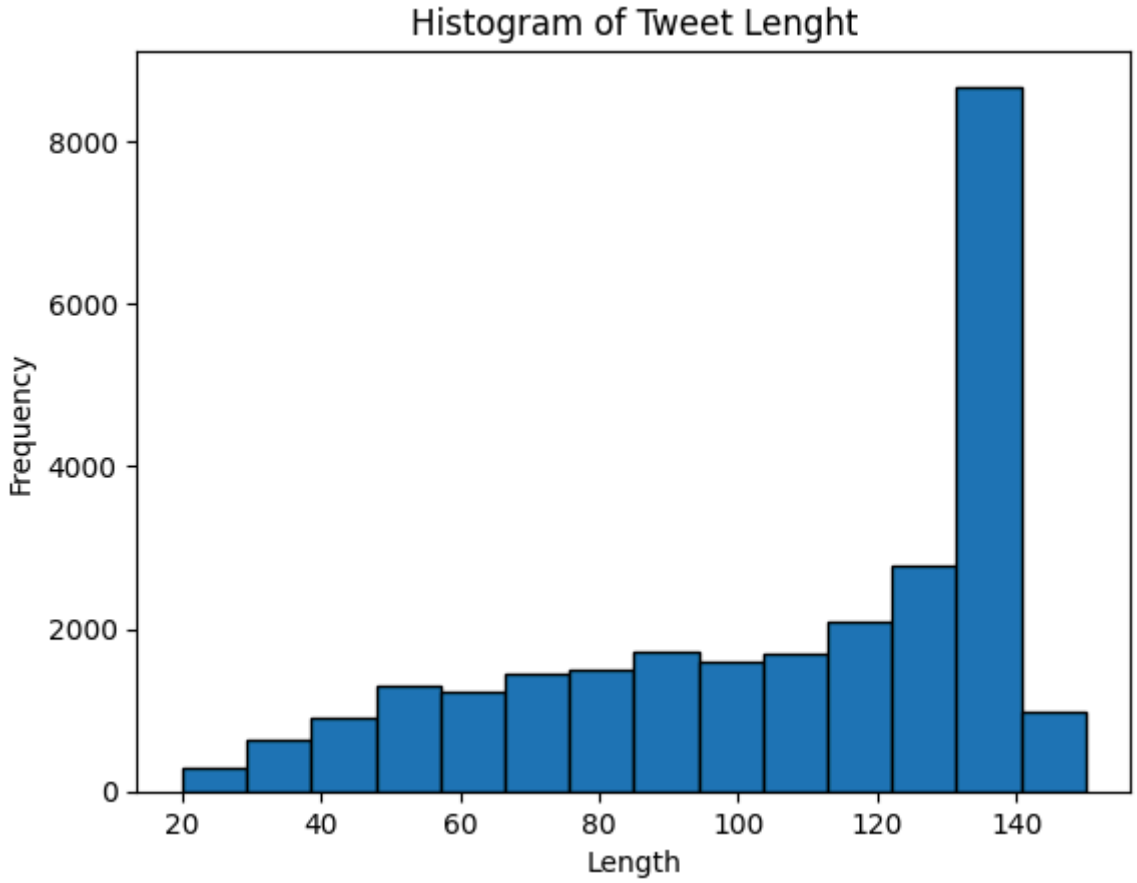
Introduction

In recent years, the explosion of social media platforms like Twitter has provided a rich source of data for analyzing public sentiment and discourse. Unsupervised clustering methods, such as K-means clustering, offer a powerful approach to explore and categorize large volumes of text data without predefined labels. This study applies K-means clustering to a corpus of tweets to uncover underlying patterns in the data, particularly focusing on the sentiment associated with different candidates from the Presidential Election of 2016. The candidates in focus here are Democratic candidate Hilary Clinton, and Republican candidate Donald Trump. Combining sentiment analysis with clustering techniques will allow us to understand how sentiment varies across different clusters of tweets and to identify key themes and sentiment trends within the corpus.

Preprocessing

The dataset I am dealing with is a collection of tweets relating to a number of important figures from the 2016 Presidential Election. These figures include Hillary Clinton, Donald Trump, Bernie Sanders, and Barack Obama. The data was collected from the website data.world and contains a multitude of columns that contains information on the polarity, subjectivity, and location of the tweets in the data set In total, there are 100,000 twitter posts from between 2016-08-30 and 2017-02-28. Initial preprocessing of the data includes the filtering out of all tweets that are not in the English Language, followed by filtering out of tweets that are related to candidates that are not Hillery Clinton or Donald Trump. Null values in the tweet_text column were filtered out, and any tweets containing links were also removed from the data set. Finally, the columns 'retweeted_status_id', 'state', 'tw_user_id', 'latitude', 'longitude', 'retweet_count',

'favorite_count', and 'device' were removed from the data set because they contain information that does not pertain to this project. This leaves us with a corpus of 26,821 tweets, with 20,238 of those tweets pertaining to Donald Trump and 6,583 of those tweets pertaining to Hillary Clinton. Since these are tweets from 2016, the length of each document in our corpus is limited to the character limit of tweets at the time, which varied based on the twitter membership of the tweet author, but should generally cap out at 140 characters. The Graph below displays a distribution of document lengths in our corpus.



I follow this up by performing sentiment analysis on the tweets in the corpus. I do this by downloading the VADER lexicon and then using VADERS in built sentiment analysis function to classify negative and positive sentiments. I classify tweets that have a sentiment score of more

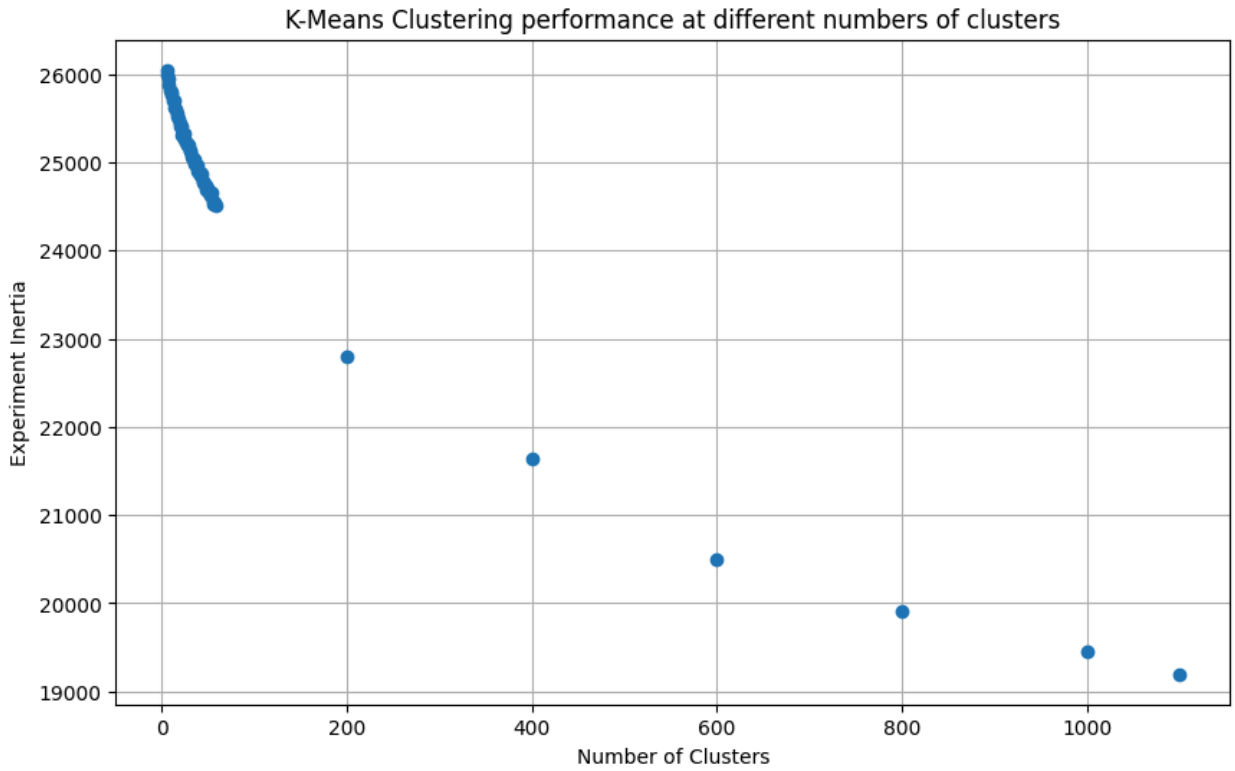
than 0.05 as positive, and tweets with a sentiment score of less than -0.05 as negative. Everything left in the middle of these values is classified as neutral.

The next stage in preprocessing involved tokenizing the word chunks in our documents, splitting into a list or ‘bag’ of words. A stop word list downloaded using the NLTK package was used to filter out words from each of the documents, and lemmatization was applied to tokens in the document to reduce them to their root form. Additionally, The punctuations, non alphabetic words, and words smaller than 4 characters in length were removed. The remaining words were converted to lower case, and the documents were stitched back together, leaving us with a processed corpus that is ready to be vectorised.

We will be using a k means clustering algorithm to group our corpus. Methods using this process to cluster twitter data have been used before (Alnajran & et.al, 2017). Hard clustering methods via k means clustering has been shown to work well with tweet style, short documents, therefore I aim to implement such a technique on the described corpus.

Term Frequency-Inverse Term Frequency scores were used to perform a K means clustering of all the documents in our corpus. Within-Cluster Sum of Squares or Inertia scores, an intrinsic performance metric, will be used to judge the clustering quality of different configurations. WCSS measures the distance of each document within a cluster from the centroid of said cluster. A good cluster will have a small WCSS value, while a less useful cluster will have larger distances between documents and the centroid, and therefore will have a bigger WCSS value. Adding the WCSS values of each cluster together will give us an inertia score for the entire clustered corpus, that will reflect the “tightness” of the performed clustering. It is worth noting that inertia will keep decreasing as cluster number increases. The theoretical minimum value a clustering inertia can have is 0. However, for a corpus with non identical

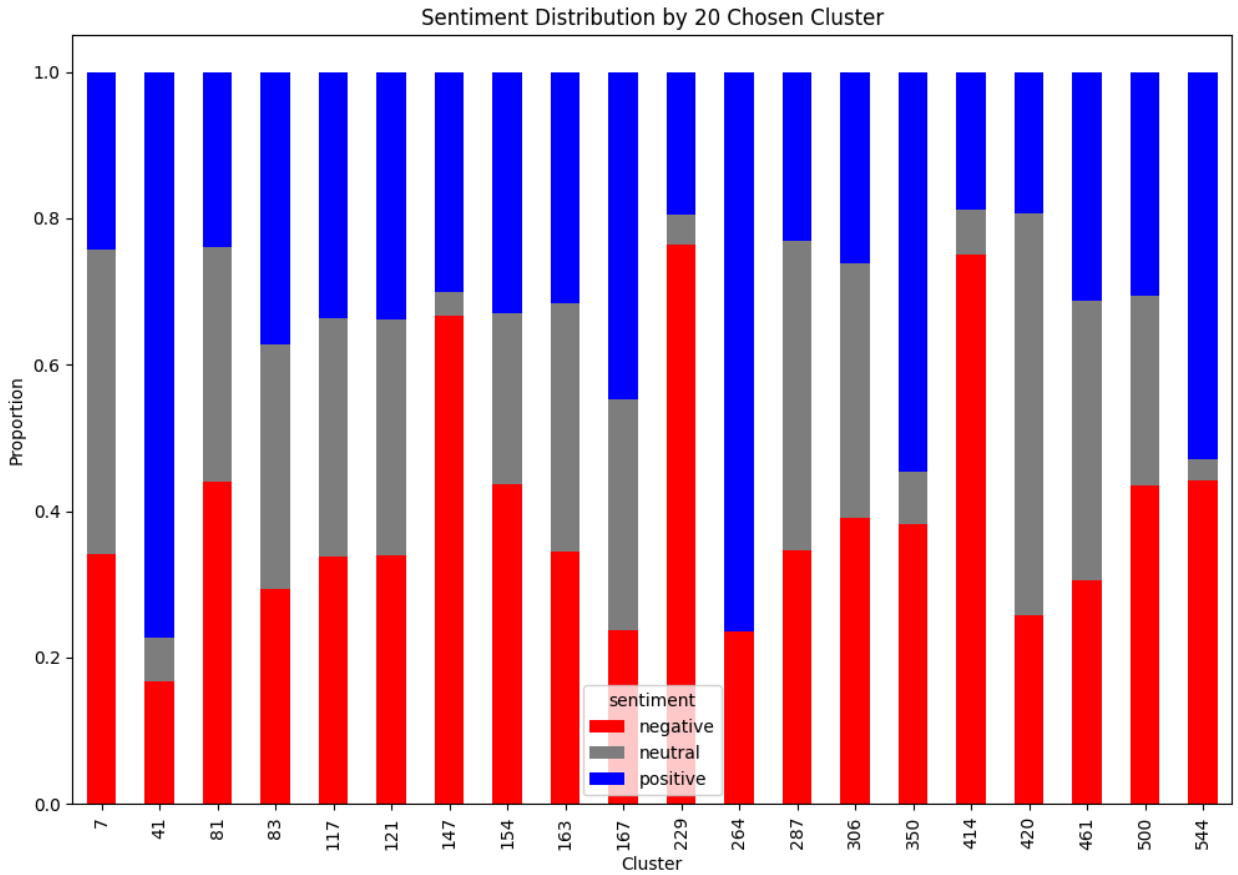
documents, such as ours, this will only happen when the number of clusters equals the number of documents in our corpus (thereby allowing each centroid to be placed exactly on each document). A set up such as this provides us with no value since it tells us nothing about the themes in our corpus. Therefore, the lowest inertia experiment will not necessarily correspond with the best k-means cluster setup. Instead, I will be looking for the "elbow point" in our experiments. This is the cluster number k where increasing the number of clusters from k to k+1 results in a significantly smaller decrease in inertia compared to the decrease observed when increasing the cluster number from k−1 to k. To do this, I will cluster the described corpus using the TF-IDF scores and measure the clustering inertia for different cluster numbers. 60 experiments were performed with cluster numbers ranging from 5 all the way up to 1100. A graph of the number of clusters versus total inertia of each cluster configuration can be found below.



From the graph, it was ascertained that the elbow point is located at the 600 cluster mark. This will be our cluster number k for our clustering experiments.

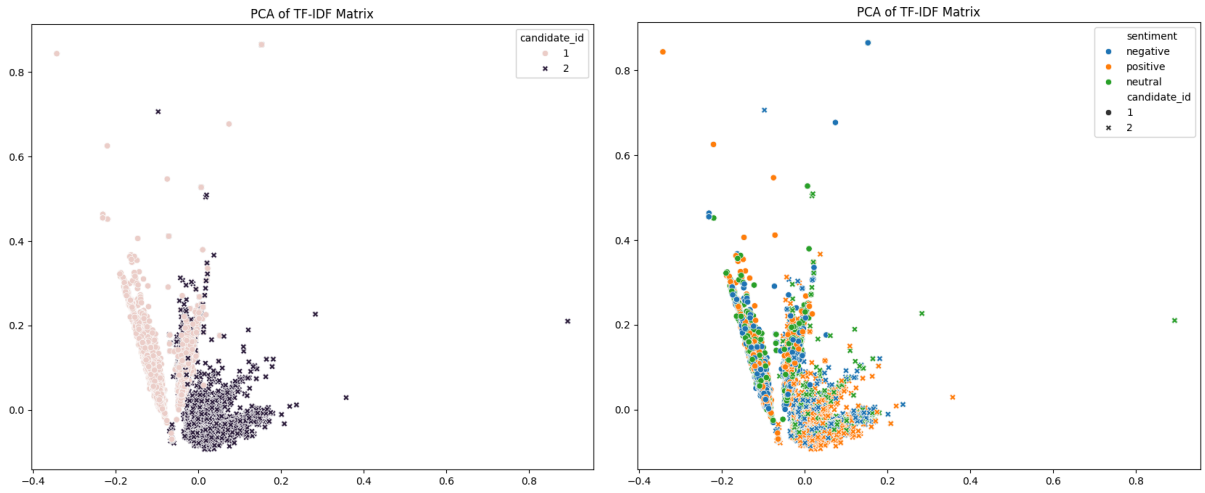
Results

Running the VADER sentiment analysis, we find that 10,255 tweets with a negative sentiment and 9,660 tweets with a positive sentiment. The remaining 6,906 tweets have a neutral sentiment. Performing the k means clustering analysis on this will then allow us to have a point of comparison between clusters based on the sentiment score. Given that the 600 clusters present are impossible to represent in a single figure. I used a random state to select for 20 clusters and graphed the cumulative sentiment scores of all of the clusters. This graph can be found below.



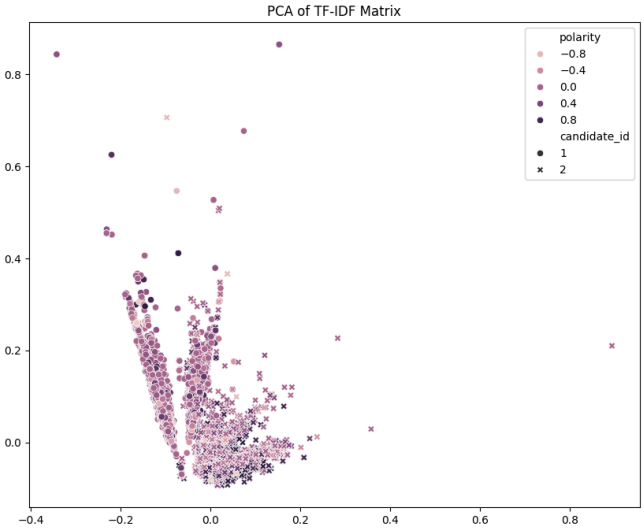
Sentiment does appear to play a significant role in clustering the corpus, with a fair degree of variance in sentiment visible across all the displayed clusters. This trend holds when inspecting all 600 of the clusters as well.

After performing the k-means clustering, I performed dimensionality reduction using a 2 dimensional PCA decomposition. This allows us to visually inspect the clustered documents. Due to the relatively large number of clusters present in this model, the PCA visualization of the clusters contains a fair amount of overlap between clusters. However, distinguishing the plotted documents by subject candidate, broad groupings in the PCA based on candidate id become visible.



Candidate 2, or Donald Trump, tweets are clustered close together near the lower center of the dimensionally reduced space, whereas candidate 1, or Hillary Clinton, tweets fan outwards in two streaks from this region. Comparing the sentiment coloring of the described areas on the graph on the left, we can gain some understanding of the perception twitter users had of these two candidates. The crosses (representing Donald Trump tweets) have a strong blend of positive and negative sentiment tweets, as pictured by the significant presence of orange and blue colors. Similarly, Hillary Clinton tweets seem to be colored more green and blue, showing that

sentiment towards her on twitter skewed more towards the neutral and negative side. This would seem to imply that Trump was a more polarizing figure on the site, since sentiments towards him were less likely to be neutral, whereas Hillary Clinton had a more consistent reception, in general. To verify this we can overlay the provided polarity scores over the PCA decomposition of the clustering groups, along with Candidate identifying markers.



The darker purple section at the lower middle of the space indicates that a significant number of Trump centric tweets are highly polarized. Clinton centric tweets on the other hand seem to consistently be near the 0.4 polarity mark, with outliers going down past 0 and up to 0.8.

Conclusion

The clustering results, coupled with sentiment analysis, demonstrate that sentiment plays a crucial role in organizing tweets into distinct groups, with observable variance across clusters. The use of dimensionality reduction techniques like PCA further allows for a visual inspection of these clusters, highlighting the polarization in sentiment towards specific candidates. The findings suggest that while sentiment towards some candidates, such as Donald Trump, tends to

be highly polarized, others, like Hillary Clinton, receive more consistently neutral or negative sentiment.

References

Alnajran, N., & et.al. (2017). *Cluster Analysis of Twitter Data: A Review of Algorithms*. e-space. Retrieved August 24, 2024, from https://e-space.mmu.ac.uk/617901/1/ICAART_2017_110_CR_final.pdf