Toronto Metropolitan University

Solving a Practical Clustering Problem

Exploring the Daily Kos Dataset

Deandra Spike-Madden 500946801

CPS 803 Machine Learning

Dr. Elodie Lugez

December 08, 2022

Background

The Bag of Words dataset used in this assignment was found in the UCI Machine Learning Repository under the default task clustering. From the data folder, I chose to work with Daily Kos blog entries. Daily Kos is a political blog and forum dedicated to discussing the American Democratic Party and liberal American politics. Founded by Markos Moulitsas in 2002, the website provides deep analysis, news and up to date information about ongoing politics. The vocabulary contains words such as “voters”, “elections”, and “bush”, which implies that some posts could be referring to an American election. The UCI Machine Learning repository provides two files: the bag of words file in sparse format and the vocabulary. The repository's sample dataset consists of 3420 documents, a vocabulary of 6906 terms, and 467,714 words. The creation of the vocabulary was based on the tokenization and elimination of stop words from each document. If the token occurred more than ten times, it was added to the vocab. Due to copyright concerns, none of the documents in the collection have any attachments or personally identifying information. Since the default task is clustering, there are no labels.

Methods

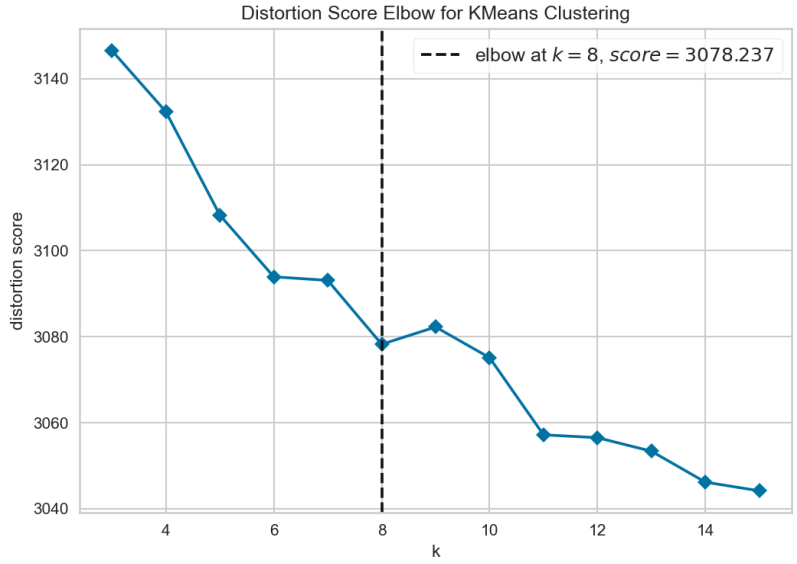
The pipeline for preprocessing the data includes building each post by using the bag of words (docword.kos.txt) and cleaning the content. There were a lot of bi-grams in the vocabulary joined together by underscores. These underscores were replaced with spaces. I eliminated words from the document that were heavily influenced by numbers and stemmed the vocabulary in an effort to lessen inertia. The Porter Stemming Algorithm was applied to each word. It works by removing the suffixes from an English word and obtaining its stem. Lemmatizing the vocabulary would not help reduce redundancy in each document due to it keeping several variations of a word. Once the lexicon was cleaned, I began to reconstruct each post. Since the feature is texted based, it was necessary to vectorize each post to get its numerical representation. This was done using the TF-IDF vectorizer by sklearn which converts documents into a matrix of TF-IDF features.

$$term\ frequency\ (tf)\ =\ count\ of\ term\ in\ document/\ total\ words\ in\ document$$
$$document\ frequency\ (df)\ =\ occurrence\ of\ term\ in\ document$$
$$inverse\ document\ frequency\ (idf)\ =\ log(N/(df\ +\ 1))$$
$$tf\ -\ idf(term, document)\ =\ tf(t,d)\ \times\ idf(t)$$

After vectorizing the text, the data type is a sparse matrix. Each row in the matrix refers to a document and each column represents a token in the vocabulary making the size of the matrix very large. To reduce the dimensionality down to three, the principal component analysis (PCA) technique was used. The model’s training process accelerated as a result. To cluster the bag of words, the KMeans algorithm was applied. This algorithm divides a dataset into k non-overlapping clusters. To choose the k value, I used the Elbow method (package) which calculated the Sum of Squared Errors (SSE) for k values ranging from 3 to 16.

$$Sum\ of\ Squared\ Errors\ (SSE)\ =\ \sum_{i=1}^{K}\sum_{X=C_i} dist^2(m_i, x\ ) = \sum(y'_i\ -\ y_i)^2$$

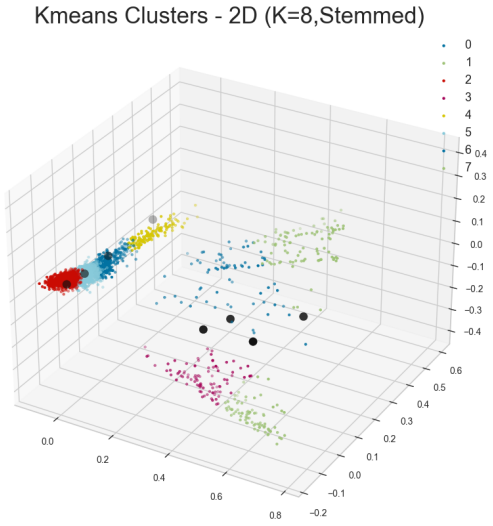
The Elbow Method suggests the optimal k value. It works by plotting the SSE against the number of clusters. The point after the SSE starts to decline linearly is known as the elbow point. When analyzing the graph from the range of k values, the objective is choose one that is low in value.



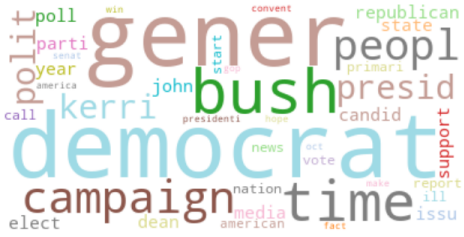
While choosing a k value for the KNN model it is important for its inertia to be low as well as the value of k itself but it is evident that as k value increases, the inertia decreases. When the inertia is low, it means the data points within each cluster are similar and separated into distinct groups. With this in mind, I decided to analyze the suggested k value 8 followed by 6 and 4.

Results

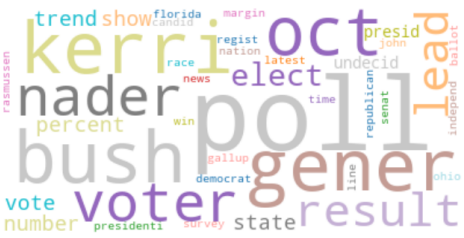
When k = 8



Inertia for model: 3087.6669559838583



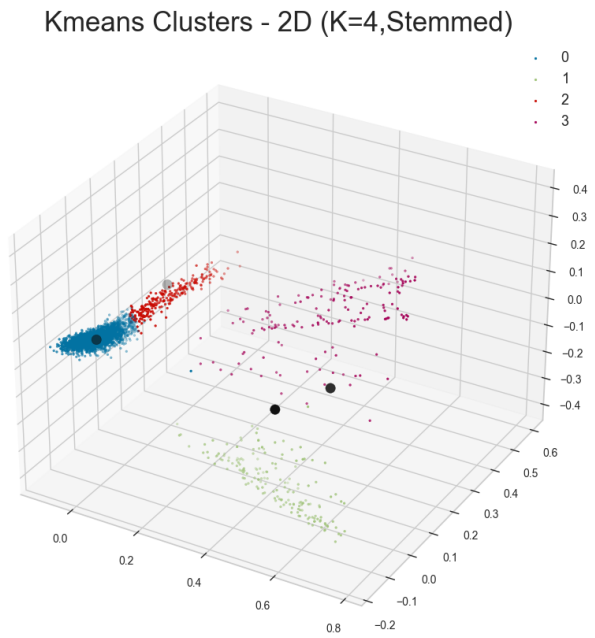
k = 8 → Cluster 5



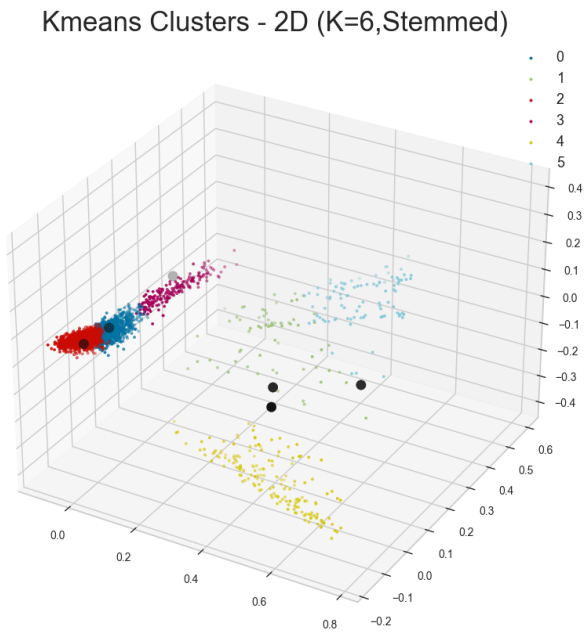
k =8 → Cluster 7

After comparing the words in each cluster, it became clear that setting k as eight was not optimal. The words in several clusters were overlapping. Certain clusters needed to be combined into one such as cluster 5 and cluster 7.

When k = 4 and k = 6



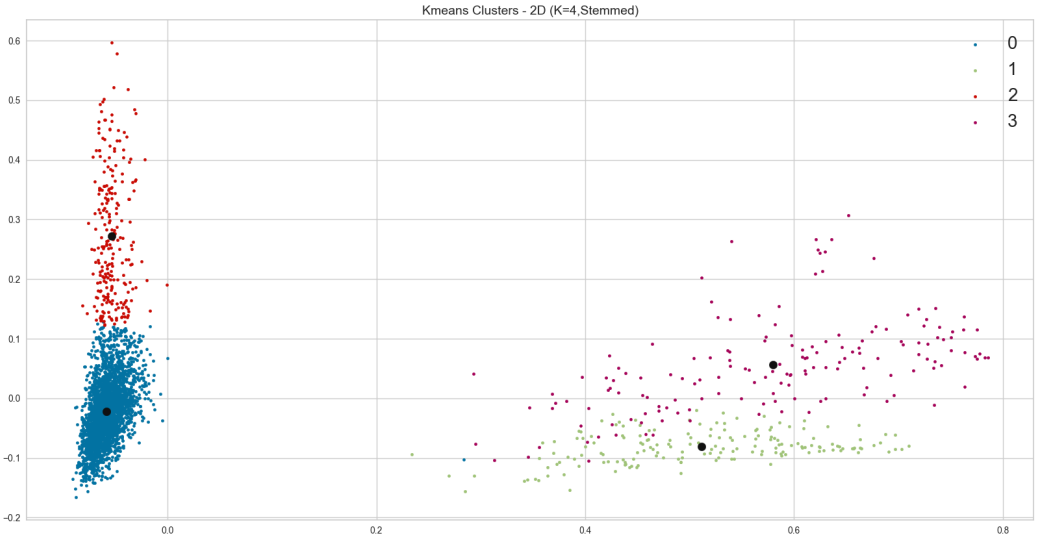
Inertia for model: 3132.3447332229607



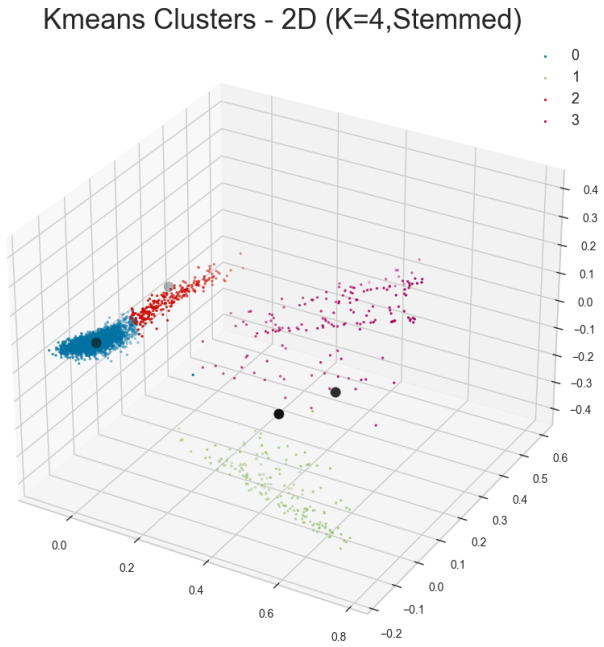
Inertia for model: 3100.4953592853662

Although the inertia is greater when k = 4, it reduces the overall amount of overlap within the clusters. Having the extra cluster to make k=6 would provide inaccurate results and cause a loss of information. In that case, information would not be insightful due to multiple clusters being very similar.

Optimal k value (k = 4)



The optimal k value for this dataset would be four. In situations where it is increased, the clusters become too small and lack insight. In the event that it is reduced, data points become generalized, and there is poor separation between each cluster.

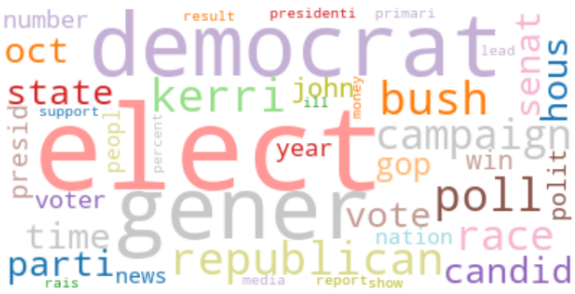


Inertia for model: 3132.3447332229607

k =4 → Cluster 0

k =4 → Cluster 1

k =4 → Cluster 2



k =4 → Cluster 3

Conclusions

When classifying features using KMeans, using the Elbow Method is helpful to determine the optimal amount of clusters but should not be the only factor considered. That method may not work for every dataset and it is important to consider other factors while deciding on a k value. By using the method as a starting point, it allows you to see what changes are needed to build a better model.

References

Franklin, S. J. (2019, November 26). *Elbow method of k-means clustering algorithm*. Medium. Retrieved December 8, 2022, from https://medium.com/analytics-vidhya/elbow-method-of-k-means-clustering-algorithm-a0c916adc540#:~:text=In%20this%20article%20we%20would%20be%20looking%20at,elbow%20method%20holds%20for%20any%20multivariate%20data%20set.