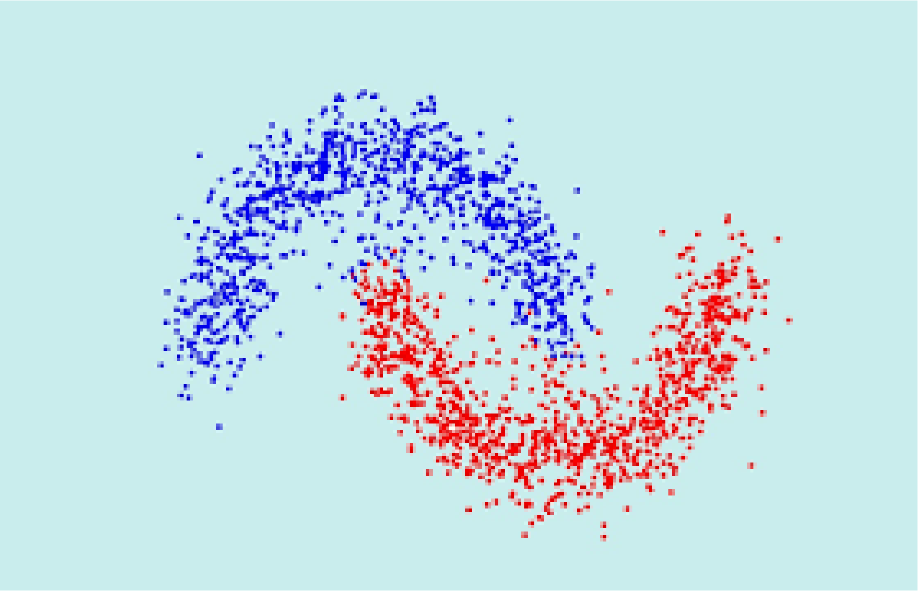
clustering



Amit Fallach

/https://www.linkedin.com/in/amitfallach

amitfallach (Amit Fallach) · GitHub

Abstract

This scientific report delves into the clustering analysis of 3,430 articles or blogs published in Kos Daily during the 2004 United States presidential election. Employing K-means and Hierarchical Clustering - Agglomerative Clustering, the study seeks to unravel hidden patterns in the political discourse. The K-means algorithm, tested with varying division methods, yielded a pivotal decision to form three clusters based on a comprehensive evaluation of silhouette scores and Sum of Squared Errors (SSE). Meanwhile, Hierarchical Clustering, utilizing a bottom-up approach with diverse linkage methods, showcased clear boundaries among clusters, with the exception of the single linkage method, which exhibited sensitivity to small differences between data points. This report presents an insightful comparative analysis of the effectiveness of these clustering algorithms in revealing underlying structures within the dataset, contributing to the broader understanding of political discourse during the 2004 US presidential election.

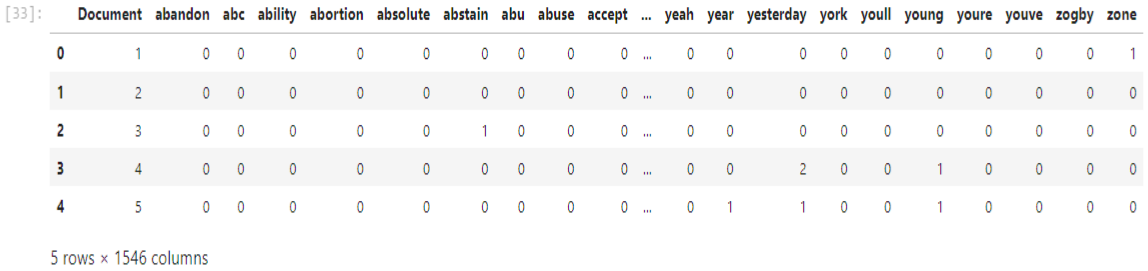
Introduction

The application of clustering algorithms, specifically k-means and Hierarchical Clustering - Agglomerative Clustering, plays a pivotal role in organizing and categorizing vast amounts of textual information. Document clustering, an established practice in information retrieval, has found widespread application in various search engines, such as PolyMeta, Helioid, and the gov.FirstGov portal, streamlining the accessibility of relevant information. In this study, we focus on applying these clustering techniques to 3,430 articles from Kos Daily, a prominent American political blog that offers a progressive perspective. Originating from the period leading to the 2004 United States presidential election, these articles encapsulate diverse viewpoints, emphasizing foreign policy, particularly the ramifications of the Iraq invasion in 2003.

The central inquiry of this investigation is twofold: firstly, to utilize the k-means and Hierarchical Clustering - Agglomerative Clustering algorithms to discern meaningful structures within the corpus of articles from Kos Daily; secondly, to evaluate the efficacy of these clustering techniques in uncovering nuanced narratives surrounding the 2004 US presidential election. Additionally, a significant part of the research objective is to delve into the inner workings of these algorithms, gaining insights into their mechanisms and behavior, as outlined in the task provided.

Data Overview

Before delving into the detailed analysis and application of clustering algorithms, it's essential to provide an overview of the structure and content of the original data frame. The screenshot below presents a snapshot of selected columns, showcasing the data types and initial entries. Understanding the data structure is fundamental to interpreting the results and conclusions drawn from the subsequent clustering analyses.



[33]:

	Document	abandon	abc	ability	abortion	absolute	abstain	abu	abuse	accept	...	yeah	year	yesterday	york	youll	young	youre	youve	zogby	zone
0	1	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	1
1	2	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
2	3	0	0	0	0	0	1	0	0	0	...	0	0	0	0	0	0	0	0	0	0
3	4	0	0	0	0	0	0	0	0	0	...	0	0	2	0	0	1	0	0	0	0
4	5	0	0	0	0	0	0	0	0	0	...	0	1	1	0	0	1	0	0	0	0

5 rows × 1546 columns

Figure 1: Snapshot of Selected Columns in the Original Data Frame.

The columns displayed include essential information about the frequency of words in each document, offering a glimpse into the variables that form the basis of the clustering algorithms. This visual representation sets the stage for a comprehensive exploration of the clustering results and the subsequent interpretation of the clustered articles from Kos Daily.

unusual things should be noted, the first is that the "document" column is a built-in index 2 column and in order not to create confusion in the division into clusters we will remove it .from the file

In addition, it can be seen that all the data are positive and have a common standard, so there is no need to perform scaling.

K-means Algorithm

K-means is a widely used clustering algorithm designed to partition a dataset into distinct groups based on similarities among data points. The "K" in K-means represents the predetermined number of clusters the algorithm aims to identify within the data. The algorithm iteratively assigns data points to the nearest cluster centroid and updates the centroid based on the mean of the assigned points. This process continues until convergence, resulting in well-defined clusters that encapsulate similar patterns or characteristics in the dataset. K-means is known for its simplicity, efficiency, and effectiveness in uncovering inherent structures within data.

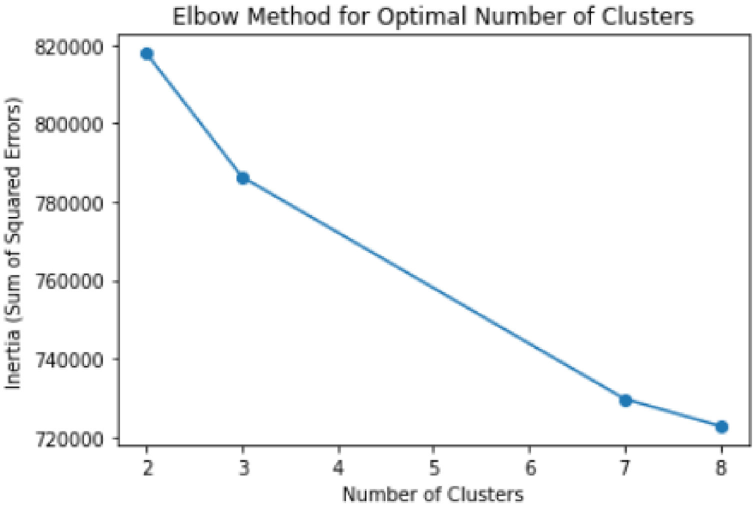
Methodology

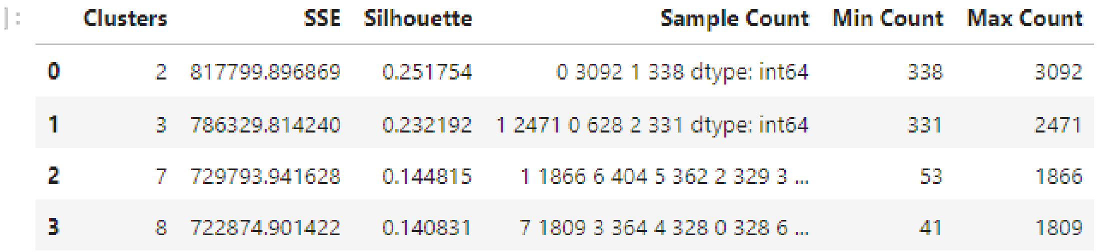
For the K-means algorithm, few division methods (2-10) were tested. The algorithm calculates the Sum of Squared Errors (SSE) and silhouette for each method.

When evaluating clustering models, the Silhouette Score serves as a valuable metric if the primary objective is to assess the separation and distinctiveness of clusters. Higher Silhouette Scores indicate well-defined and separated clusters. On the other hand, if the focus is on determining the optimal number of clusters and assessing their compactness, the Sum of Squared Errors (SSE) becomes a useful metric. Particularly when employed in conjunction with the elbow method, SSE helps in identifying a suitable number of clusters by measuring the compactness of the clusters.

Work process and results

A function was developed to implement the algorithm for each division method, providing the required parameters. Results were presented in an ordered data frame, and an elbow method graph was used to find the optimal K according to SSE.





	Clusters	SSE	Silhouette	Sample Count	Min Count	Max Count
0	2	817799.896869	0.251754	0 3092 1 338 dtype: int64	338	3092
1	3	786329.814240	0.232192	1 2471 0 628 2 331 dtype: int64	331	2471
2	7	729793.941628	0.144815	1 1866 6 404 5 362 2 329 3 ...	53	1866
3	8	722874.901422	0.140831	7 1809 3 364 4 328 0 328 6 ...	41	1809

Figure 2: Elbow Method.

It can be seen that the SSE values are high and the silhouette is very low, in addition, it is not possible to unequivocally decide from the graph on an optimal division into clusters, therefore we will use scaled inertia.

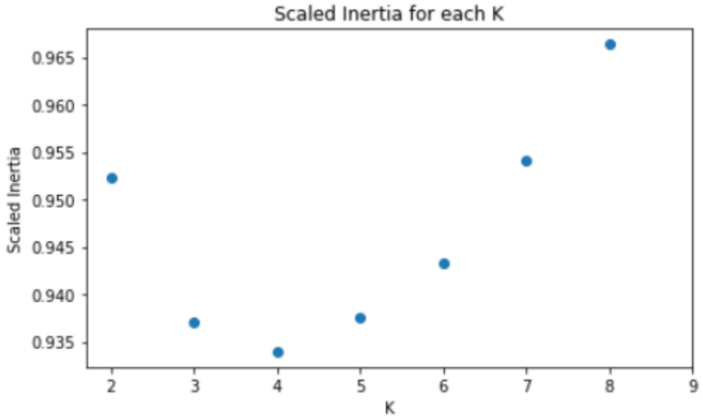
Work process and results

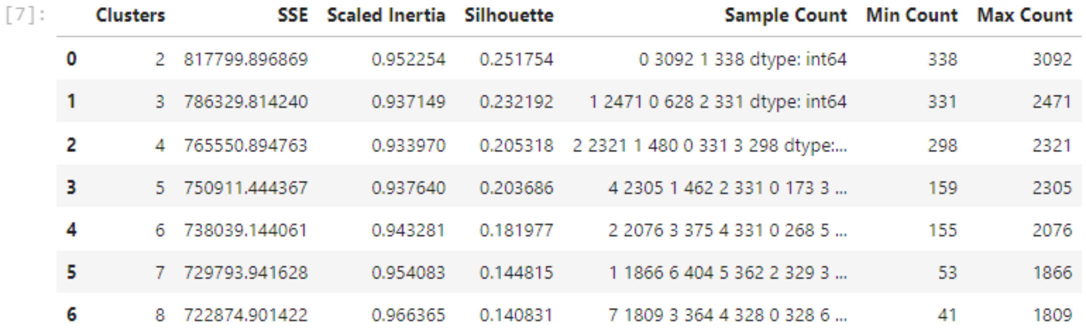
The proposed approach takes into account the inertia value for each possible K and weights it by a penalty parameter. This parameter represents the trade-off between the inertia and the number of clusters. Instead of using inertia by itself, we calculated a weighted version of it that helps find the optimum point:

$$Scaled\ Inertia = \frac{Inertia(K)}{Inertia(K=1)} + \alpha \cdot K$$

Scaled Inertia Formula

Credit: An Approach for Choosing Number of Clusters for K-Means | by Or Herman-Saffar | Towards Data Science





[7]:

	Clusters	SSE	Scaled Inertia	Silhouette	Sample Count	Min Count	Max Count
0	2	817799.896869	0.952254	0.251754	0 3092 1 338 dtype: int64	338	3092
1	3	786329.814240	0.937149	0.232192	1 2471 0 628 2 331 dtype: int64	331	2471
2	4	765550.894763	0.933970	0.205318	2 2321 1 480 0 331 3 298 dtype:...	298	2321
3	5	750911.444367	0.937640	0.203686	4 2305 1 462 2 331 0 173 3 ...	159	2305
4	6	738039.144061	0.943281	0.181977	2 2076 3 375 4 331 0 268 5 ...	155	2076
5	7	729793.941628	0.954083	0.144815	1 1866 6 404 5 362 2 329 3 ...	53	1866
6	8	722874.901422	0.966365	0.140831	7 1809 3 364 4 328 0 328 6 ...	41	1809

Figure 3: Scaled Inertia for each K.

After using Scaled Inertia and a graphical presentation, it can be seen that the optimum is reached when K=4. We will continue to check the silhouette.

In-Depth Analysis of Clusters in K-means

After checking the SSE and the scaled inertia, I saw that the optimum is reached by dividing into 4 clusters. In order to make a decision, I decided to conduct a more in-depth examination and examined the silhouette as well . Silhouette analysis for clusters ranging from 2 to 10 was visualized in a matrix. The goal was to assess the quality of each cluster division and make a nuanced decision.

Key Considerations

1. Average Silhouette Value

The silhouette plots displayed the silhouette scores for individual clusters within each - division. It was crucial to ensure that the average silhouette score exceeded a certain threshold. A higher average silhouette suggests well-defined and distinct clusters.

2. Consistent Cluster Sizes

A critical aspect of the analysis was the examination of cluster sizes. For robust - clustering, it was essential that the clusters were approximately the same size. This ensures that no single cluster dominates the others, allowing for a balanced representation of distinct patterns within the data.

By scrutinizing the silhouette plots and considering both the average silhouette score and the uniformity of cluster sizes, a comprehensive understanding of the optimal cluster division was achieved. These additional insights helped in making an informed decision regarding the number of clusters that best represented the underlying structure of the dataset.

Silhouette analysis

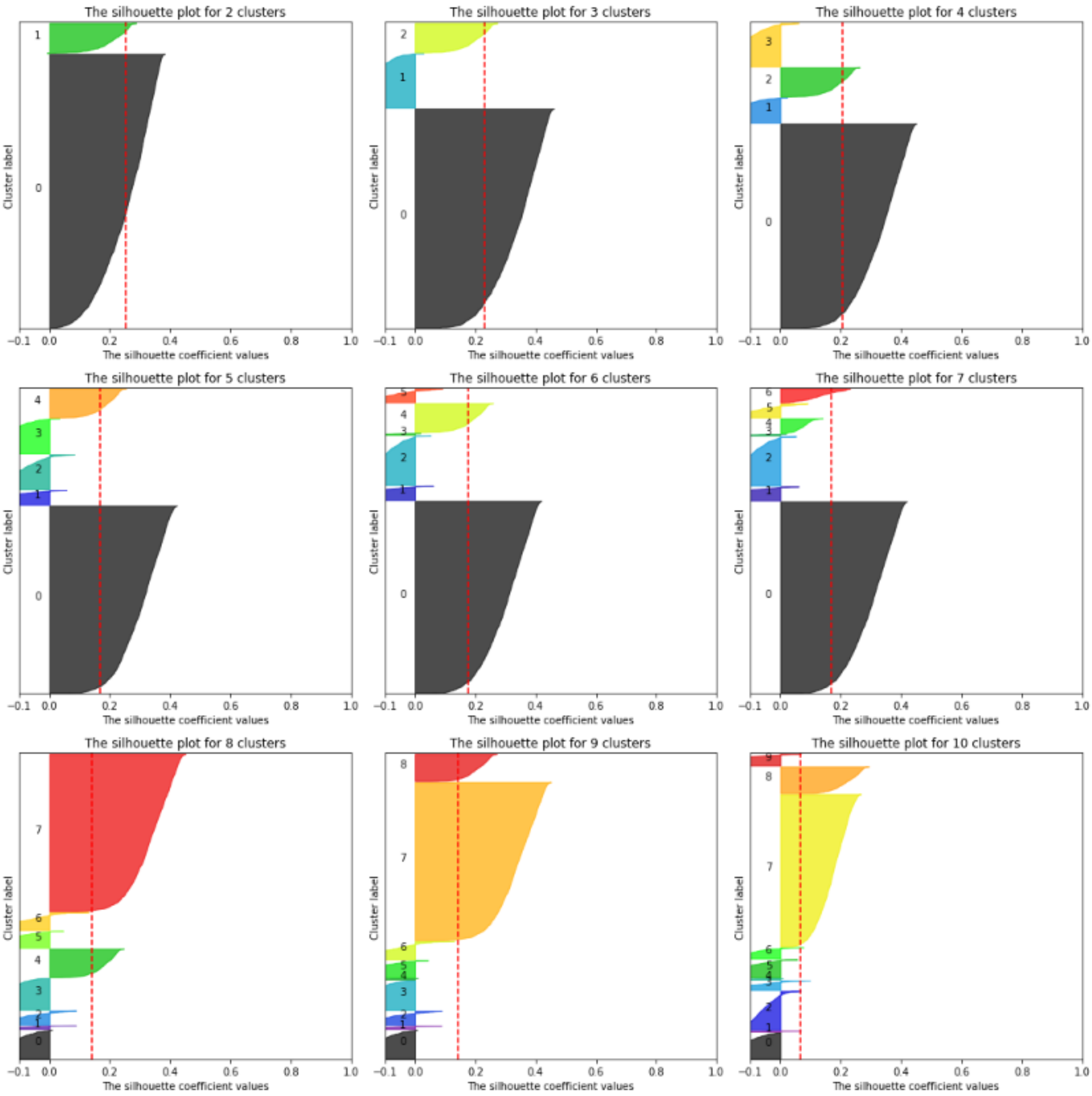


Figure 4: In-Depth Silhouette Analysis.

It can be seen that most of the divisions have clusters that overlap or almost overlap (we can learn from the graph and the low silhouette score).

All the tests done so far did not bring up a definite unequivocal answer, so I built a function that outputs the most frequent words in each cluster. Later we will try to examine in the leading divisions the leading words in each cluster to see if it is clear what characterizes it.

Currently the top divisions are 2 (positive silhouette graph) and 4 (low scaled inertia scores).

Cluster Analysis

I divided the data separately into 2 clusters and 4 clusters, each time I used a function that exports the 30 most frequent words in each cluster.

The goal is to understand if there is a central theme that revolves around the leading words in each cluster.

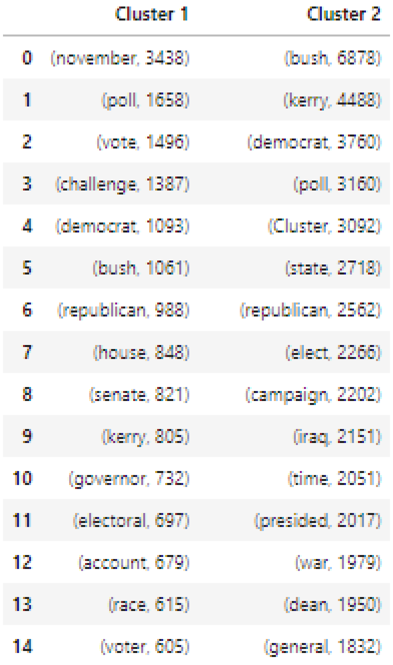
K=2

Cluster 1:

the first list of words centers around the broader electoral process, evident in the presence of terms like "november," "vote," and "challenge." The unique addition of words such as "labor" and "scoop" suggests a focus on specific events or challenges within the electoral context. Unlike the second list, **the first list captures a more general perspective on elections, encompassing a wider array of topics beyond the specific context of the Iraq War.**

Cluster 2:

The second list of words, characterized by terms like "Iraq," "war," and "American," is indicative of a thematic emphasis on the Iraq War and its connection to the broader landscape of American politics. Notably, the inclusion of words such as "Kerry," "Republican," and "democrat" aligns with the timeframe of the 2004 U.S. presidential election, **highlighting a narrative that intertwines the political landscape with the ongoing war.**



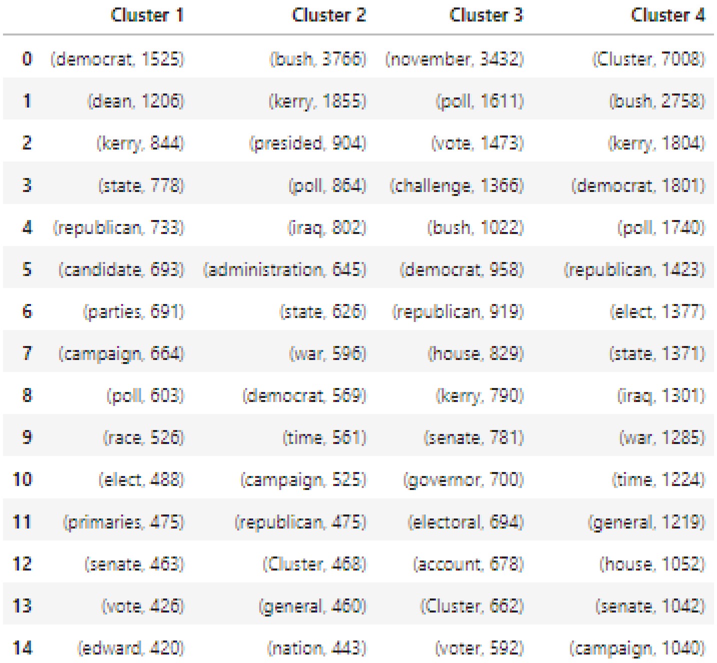
	Cluster 1	Cluster 2
0	(november, 3438)	(bush, 6878)
1	(poll, 1658)	(kerry, 4488)
2	(vote, 1496)	(democrat, 3760)
3	(challenge, 1387)	(poll, 3160)
4	(democrat, 1093)	(Cluster, 3092)
5	(bush, 1061)	(state, 2718)
6	(republican, 988)	(republican, 2562)
7	(house, 848)	(elect, 2266)
8	(senate, 821)	(campaign, 2202)
9	(kerry, 805)	(iraq, 2151)
10	(governor, 732)	(time, 2051)
11	(electoral, 697)	(presided, 2017)
12	(account, 679)	(war, 1979)
13	(race, 615)	(dean, 1950)
14	(voter, 605)	(general, 1832)

Figure 5: top words for K=2.

Cluster Analysis

K=4

In all the clusters, words related to the preliminary elections and the elections, the war in Iraq and other common topics were repeated so that it was not possible to choose a topic that characterizes each cluster.



	Cluster 1	Cluster 2	Cluster 3	Cluster 4
0	(democrat, 1525)	(bush, 3766)	(november, 3432)	(Cluster, 7008)
1	(dean, 1206)	(kerry, 1855)	(poll, 1611)	(bush, 2758)
2	(kerry, 844)	(presided, 904)	(vote, 1473)	(kerry, 1804)
3	(state, 778)	(poll, 864)	(challenge, 1366)	(democrat, 1801)
4	(republican, 733)	(iraq, 802)	(bush, 1022)	(poll, 1740)
5	(candidate, 693)	(administration, 645)	(democrat, 958)	(republican, 1423)
6	(parties, 691)	(state, 626)	(republican, 919)	(elect, 1377)
7	(campaign, 664)	(war, 596)	(house, 829)	(state, 1371)
8	(poll, 603)	(democrat, 569)	(kerry, 790)	(iraq, 1301)
9	(race, 526)	(time, 561)	(senate, 781)	(war, 1285)
10	(elect, 488)	(campaign, 525)	(governor, 700)	(time, 1224)
11	(primaries, 475)	(republican, 475)	(electoral, 694)	(general, 1219)
12	(senate, 463)	(Cluster, 468)	(account, 678)	(house, 1052)
13	(vote, 426)	(general, 460)	(Cluster, 662)	(senate, 1042)
14	(edward, 420)	(nation, 443)	(voter, 592)	(campaign, 1040)

Figure 6: top words for K=4.

discussion and conclusions:

After a detailed examination of the silhouette and thorough investigation of the SSE (scaled inertia), I considered two possible approaches to clustering—either with 2 clusters or 4 clusters. While the 4-cluster division yielded an optimal scaled inertia, concerns arose during the silhouette test regarding potential overlapping clusters. Furthermore, upon examining the leading words in each cluster for the 4-cluster division, the conclusions regarding the subject that characterizes each cluster were not unequivocal. **As a result, I opted to explore the division into 2 clusters due to the clearer and more coherent patterns observed in this approach.**

Hierarchical Clustering - Agglomerative Clustering

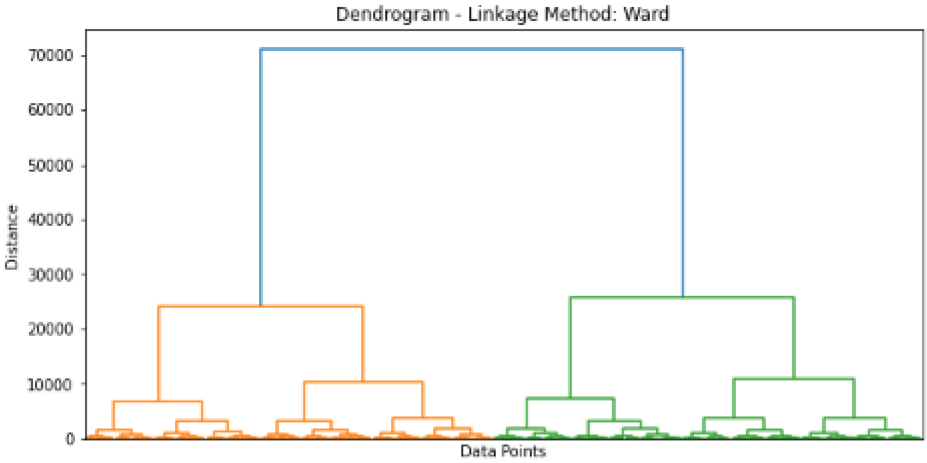
Agglomerative Clustering is a hierarchical clustering algorithm that begins with each data point as an individual cluster and iteratively merges the closest clusters until a single cluster encapsulates all data points. This process creates a tree-like structure, known as a dendrogram, representing the hierarchy of relationships between clusters. Agglomerative Clustering is characterized by its flexibility, allowing for the identification of clusters at various levels of granularity. This method is particularly useful for revealing hierarchical structures within datasets, providing a visual representation of how data points are progressively grouped into clusters.

Methodology

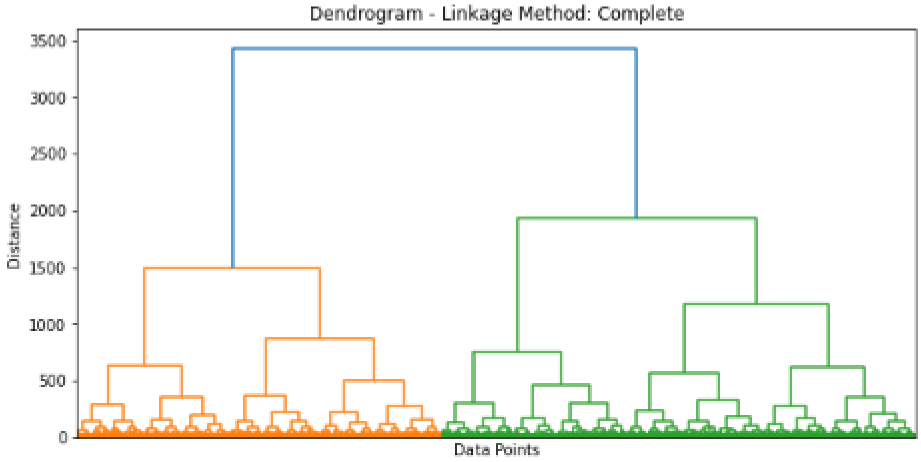
A bottom-up approach was employed for Hierarchical Clustering, specifically using Ward, Maximum (complete linkage), Average linkage, and Single linkage methods.

Work process and results

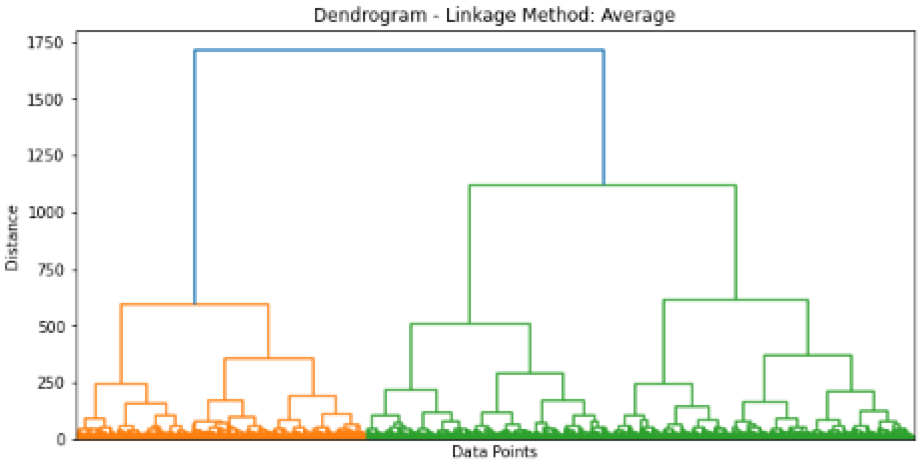
Ward Method: Minimizes the variance within each cluster, suitable for compact and - equally sized clusters.



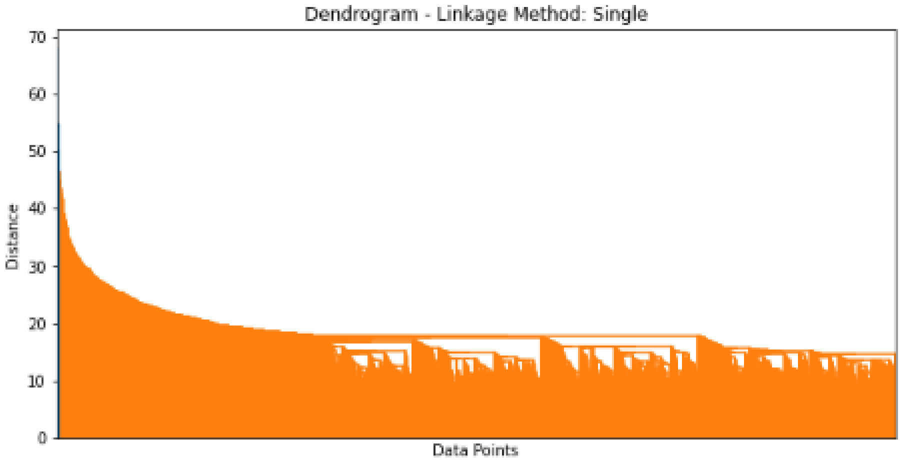
Maximum (Complete Linkage) Method: Measures dissimilarity between clusters by considering the maximum distance between their members.



Average Linkage Method: Calculates the average distance between all pairs of members in different clusters.



Single Linkage Method: Measures dissimilarity by considering the minimum distance - between clusters.



The relative interval graphs of the dendrograms for the three hierarchical clustering methods - Ward, Complete, and Average linkage, reveal distinctive characteristics. In the Ward method, the lower part of the tree exhibits relatively high cluster connections, with a slightly higher prevalence of mixed connections compared to the other two methods. The connections in the Ward and Complete methods appear at similar heights, creating a relatively symmetrical tree structure. However, the Average linkage method presents a different pattern, with clusters on the right side connecting at a higher point than those on the left side, introducing an asymmetry in the dendrogram. Notably, the Single linkage method produced a considerably different outcome, characterized by a chaotic arrangement of numerous connections concentrated at the bottom of the dendrogram. This disorderly structure suggests a higher sensitivity to small differences between data points, resulting in a less cohesive clustering outcome.