CSDS 435: Project 2

John Mays, "Group 13"

Due 04/04/23, Submitted 04/05/23; Dr. Li

1 Introduction

In this paper, I explore two (because I am one student) clustering methods along with two distance measures. The two clustering methods I chose were k-Medoids and Complete-Link Hierarchical Clustering, because both accepted pre-generated distance kernels. This went in line with the concurrent goal of also studying distance measures. I compare the algorithms and the distance measures by metrics of separation and cohesion, and engage in hyperparameter tuning and visualization in order to explore these methods.

2 Methods

2.1 Distance Measures

2.1.1 Normalized Unordered Edit Distance

$$d_1(x, y) = \frac{\sum_k x_k - y_k}{\sum_k x_k + y_k}$$

The intuition behind this was, when I was considering how you compare strings, I was reminded of edit distance (both for words and sentences). Of course, I realized that edit distance required an ordered string, and our bag-of-words model doesn't preserve the order. But I thought, very simply, the difference between two vectors is essentially a version of edit distance that does not pay attention to order (hence "unordered"). But I also thought two four-word sentences with two words in common are about as near as two six-word sentences with three words in common, or at least that this might be a good hypothesis to test since my other distance measure *does not* treat the data this way. Hence, I normalize the distance by dividing by the total number of words in both. This distance measure is $\frac{\text{\# of words not in common}}{\text{\# total number of words in both sentences}}$. Therfore, the minimum value ($d_1(x, x)$) is zero, and the maximum value (for vectors with no words in common) is one.

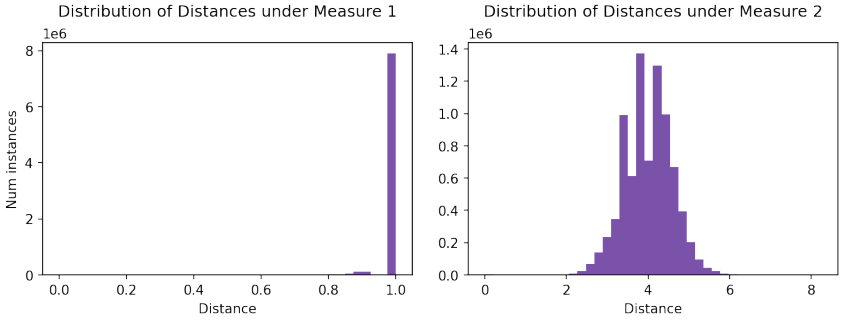
2.1.2 Euclidean Distance

$$d_2(x, y) = \sqrt{\sum_k (x_k - y_k)^2}$$

Euclidean distance seemed intuitive. It will, as opposed to my first distance measure, attribute greater distances on average to longer tweets. Two sixty-word tweets will be much farther apart than two ten-word tweets (assuming they each have nothing in common).

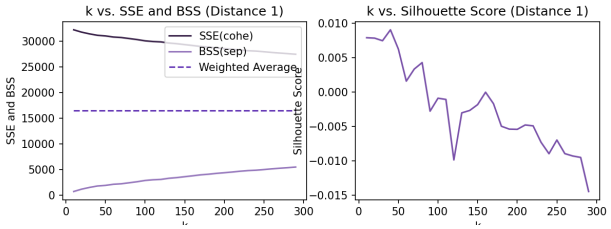
2.1.3 Comparisons

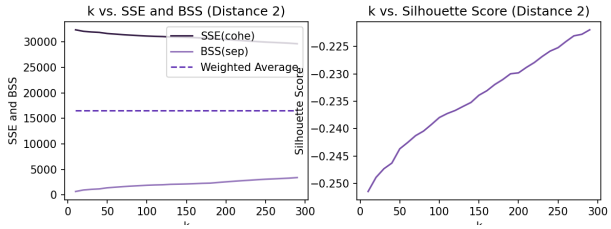
The distributions are extremely different. The Euclidean distance's (d_2) is almost normal with $\mu \approx 4$, while d_1 has an extraordinary amount of 1.0 (no words in common) and then just a few thousand distances clustered around 0.9.



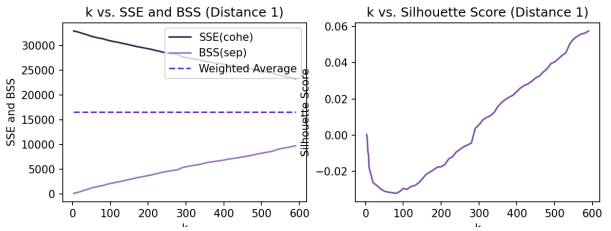
2.2 Choosing Hyperparameters

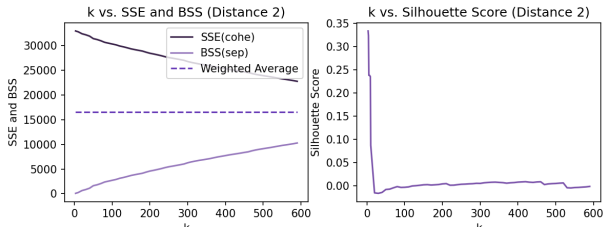
2.2.1 Choosing k for k-Medoids:





2.2.2 Choosing k for Hierarchical:





Both of my methods required me to choose k (number of clusters). In order to do this, I tried to observe how different metrics and objective functions varied with k. Some results were surprising, and revealed to me that different distances worked very differently with different clustering algorithms. Observing SSE and BSS turned out not to be very helpful, but observing the silhouette score was. In general, a silhouette score closer to one is better. The only two combinations that demonstrated a positive correlation between silhouette score and k were {Distance 1 and Hierarchical} and {Distance 2 and k-Medoids}. The other two were negatively correlated, and less so. For k-Medoids I chose a default k of 50 as that is the max silhouette achieved with either distance. And for Hierarchical, smaller choices were not giving me coherent clusters, so I chose a much higher k based on the d_1 figure: 600.

3 Data

3.1 Dataset Summary

# Num of Tweets	# Num of Words (total)	# Num of Tokens	# Avg Num of Words
4,061	32,612	9,609	8.031

Top Ten Legal Tokens:

"health", "getfit", "new", "@cnnhealth", "today's", "cancer", "know", "kids", "@drsanjaygupta", "ebola"

3.2 Stopwords:

Initially, I tried manual stopword selection, but I found this task to be cumbersome, and to leave in less frequent stopwords that were still important. So I chose most of the stopwords from the *Natural Language Toolkit* (NLTK), electing to leave a certain few out, while also trying my best to include ones specific to the dataset. After subtracting the NLTK words, I scanned through the most popular 100 words still left. The words *"rt"* (retweet) and *"w/"* (with) were removed, along with spurious characters that somehow made it through my regex cleaning.

When chosing what to keep and what to toss, I had a hard time deciding. Common stopwords like *"was"* and *"is"* contain tense information that could hypothetically be related to a cluster's meaning, but I chose to go with precedent, which suggest that they are more noise than they are worth. I elected to remove the URLs since they were almost never in common between tweets. However, I chose to leave "@" mentions and "hashtags" in. I removed the hashtag character, however, as I judged "#ebola" and "ebola" could be treated as one in the same. Hashtags and *"@"*-mentions were left in because I thought that clusters could potentially be formed around them. Quite often, a hashtag's sole purpose is to distill the subject or message of the tweet. And although not all textit"@"-mentions feature a certain doctor or expert, the ones that do and are the same are intrinsically related to one another.

4 Results

k-Medoids with d_1:

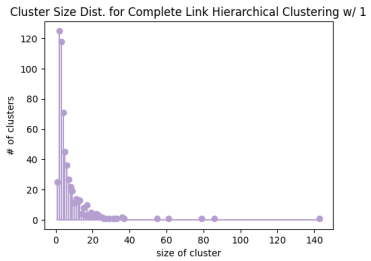
Number of Clusters	SSE	BSS	Silhouette Score
50	31061.623	1960.594	0.006

k-Medoids with d_2:

Number of Clusters	SSE	BSS	Silhouette Score
50	31631.14	1391.078	-0.244

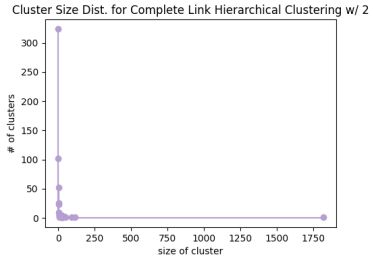
Hierarchical with d_1:

Number of Clusters	SSE	BSS	Silhouette Score
600	23151.933	9870.284	0.058



Hierarchical with d_2:

Number of Clusters	SSE	BSS	Silhouette Score
600	22643.094	10379.124	-0.001



Cluster Deciphering

For Complete-Link Hierarchical Clustering, I had what looked like the best clusters, so here some of them are:
Note: this was run with $k = 600$ and d_1.

Cluster:	Cluster 3	Cluster 1	Cluster 109	Cluster 86
size:	143	86	61	55
Popular tokens:	getfit	health	cancer	ebola
	today's	minute	breast	american
	eat	insurance	one	@jechristensen
	make	good	survival	infected
	eating	weight	scary	@elizcohencnn
	calories	crisis	prevent	ebolaqanda
	@shape_magazine	today's	new	@who
	day	loss	come	patients
	every	know	@cnnhealth	like
	healthy	food	you?	experimental

There are some really nice clusters here. **Cluster 3** has a prevailing theme of diet and weight loss. **Cluster 1** is seems to be about modern weight problems. **Cluster 109** is mostly scare-tweets about breast cancer and survival rates. **Cluster 86**

is about the ebola outbreak in America.

4.1 Cluster Visualizations:

Since there are several thousand pixels, and the details are finer, the sorted distance matrices can be found in the appendix at a large scale...

4.2 Cluster Consistency:

I used entropy an purity to compare k-Medoids to Complete Link Hierarchical Clustering, both using distance measure 2 (Euclidean). Here are my results:

- When k-Medoids is the label, Entropy = 0.822, Purity = 0.81
- When Complete Link Hierarchical Clustering is the label, Entropy = 3.164, Purity = 0.457

5 Conclusion

Even with our self-imposed limitation of only kernel methods, we managed to run a successful comparison and try a somewhat novel distance measure. Keeping the limited scope of this report in mind, at least on my two distance measures, Hierarchical clustering (with Complete Linkage) seemed to consistently outperform k-Medoids in measures of cohesion and separation, silhouette score, and common understanding. The clustering was generally much more coherent coming from Hierarchical Clustering, *especially* with distance metric one. Although this report is by no means a complete dive into clustering, let alone these two metrics, it gave us some insight into how the two methods work in limited cases, and despite having an upsetting-looking distribution at first, novel distance measure d_1 = Normalized Unordered Edit Distance proved functional and exceled against Euclidean distance.

Statement of Participation

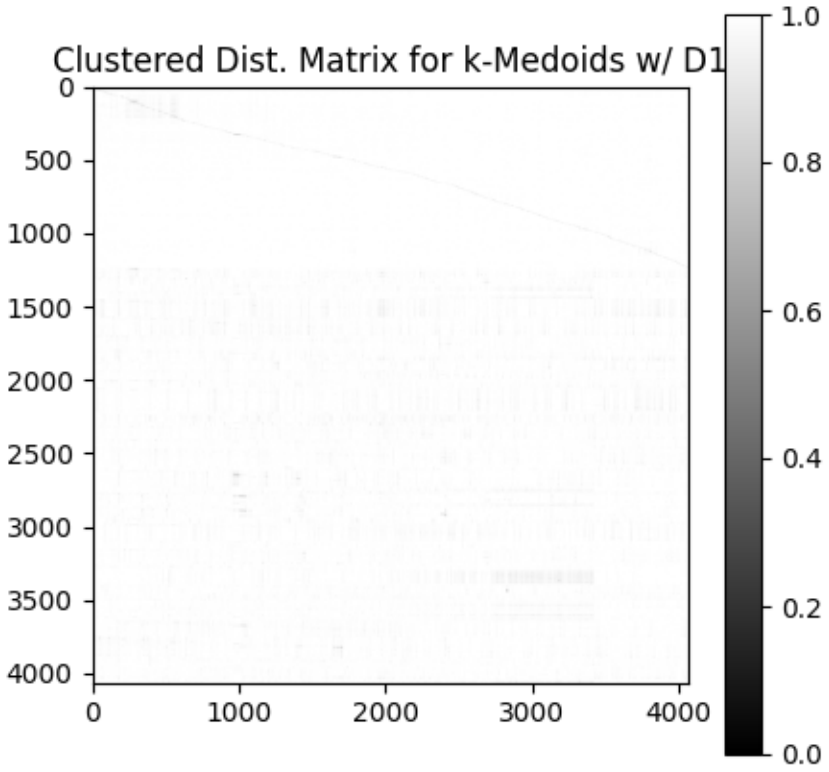
- John Mays: 100%

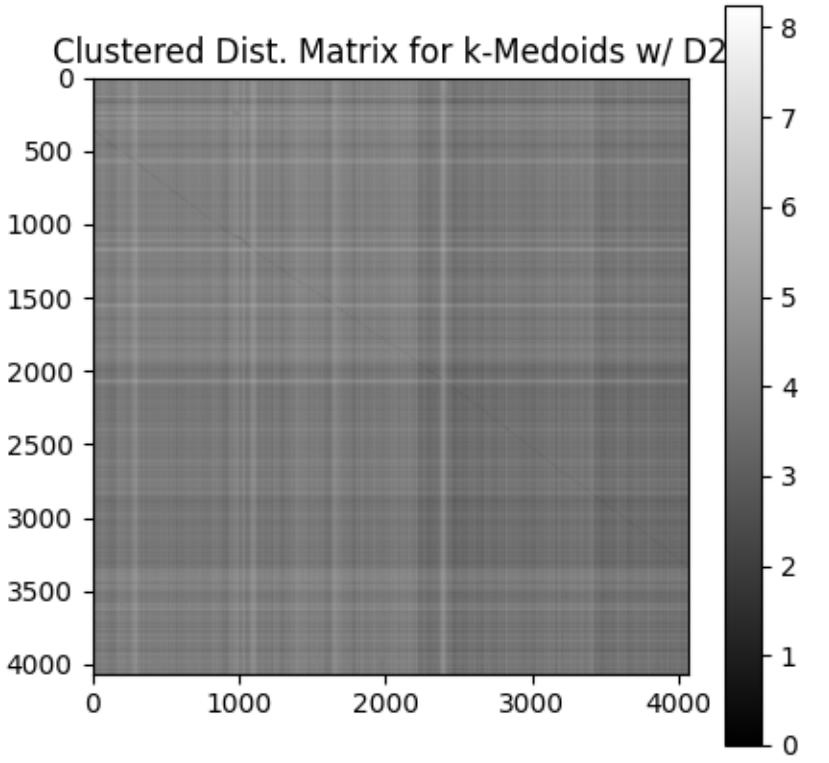
All team members agree with the specified effort.

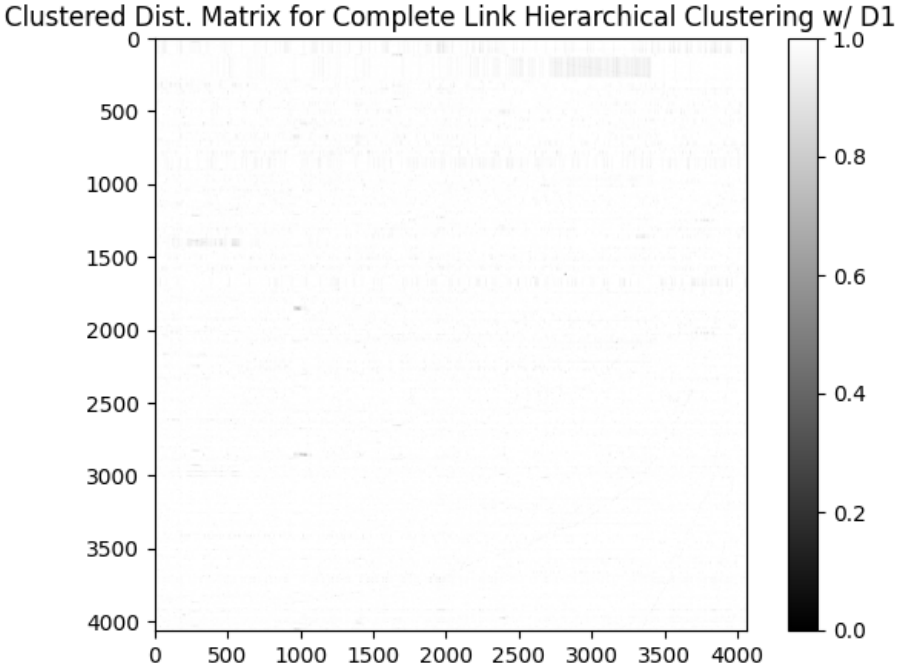
Sources

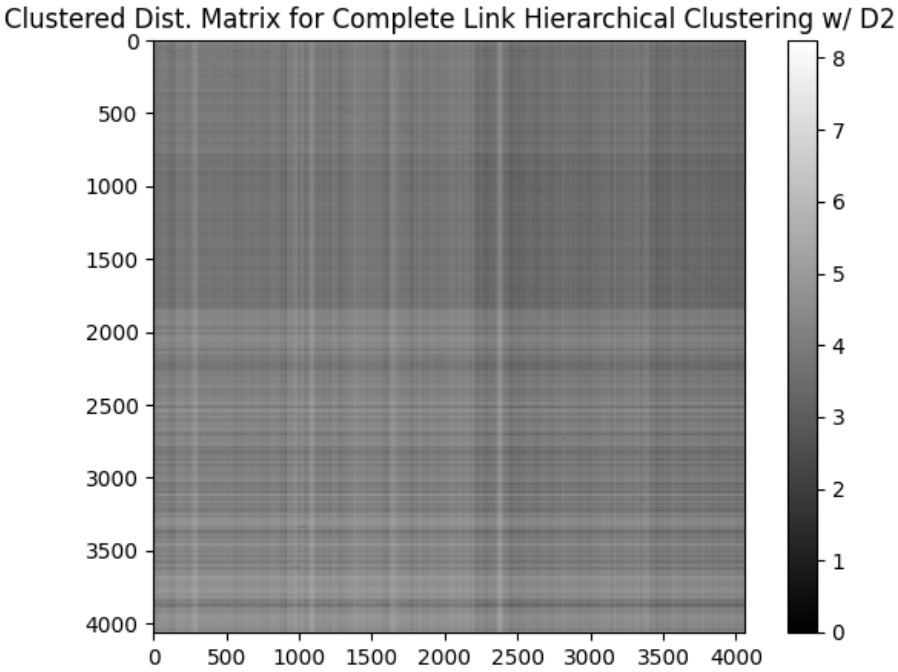
[1] *Text pre-processing: Stop words removal using different libraries(NLTK)*
[2] *Introduction to Data Mining (2nd ed.)* Tan et al.
[3] *Fuzzy Approach Topic Discovery in Health and Medical Corpora* by Karami, Gangopadhyay, Zhou, and Kharrazi

Appendix


Clustered Dist. Matrix for k-Medoids w/ D1
0
500
1000
1500
2000
2500
3000
3500
4000
0
1000
2000
3000
4000
1.0
0.8
0.6
0.4
0.2
0.0


Clustered Dist. Matrix for k-Medoids w/ D2
0
500
1000
1500
2000
2500
3000
3500
4000
0
1000
2000
3000
4000
8
7
6
5
4
3
2
1
0


Clustered Dist. Matrix for Complete Link Hierarchical Clustering w/ D1
0
500
1000
1500
2000
2500
3000
3500
4000
0
500
1000
1500
2000
2500
3000
3500
4000
1.0
0.8
0.6
0.4
0.2
0.0


Clustered Dist. Matrix for Complete Link Hierarchical Clustering w/ D2
0
500
1000
1500
2000
2500
3000
3500
4000
0
500
1000
1500
2000
2500
3000
3500
4000
8
7
6
5
4
3
2
1
0