CS 6375
ASSIGNMENT 2

Names of students in your group:
Aditya Kulkarni (axk230069@utdallas.edu)
Christopher Fan (clf170000@utdallas.edu)
Namrata Nag (nxn230019@utdallas.edu)

Number of free late days used: 0

Note: You are allowed a **total** of 4 free late days for the **entire semester**. You can use at most 2 for each assignment. After that, there will be a penalty of 10% for each late day.

Please list clearly all the sources/references that you have used in this assignment.

Solution: Repository: 6375_Assignment_2_KMeans

Report:

1. Data File Used

   From the given data, we are using "gdnhealthcare.txt" file. The file has *2997* samples.
   Format of file: <id>|<date time information>|<tweet>
   The tweets contain URLs, hashtags, and user ids.
   Note: any other data file can be used in the same code via command line argument.

2. Pre-Processing

For pre-processing the data, we have used simple string functions like replace(), split() and strip(), as well as regular expressions for removing URLs and other symbols.

i. Remove the tweet id and timestamp.

```
20          delim = lines[i].split("|")[2:]
21          lines[i] = " | ".join(delim)
```

ii. Remove any word that starts with the symbol @ e.g. @AnnaMedaris.

```
24           lines[i] = " ".join(filter(lambda x: x[0] != '@', lines[i].split()))
```

iii. Remove any hashtag symbols e.g. convert #depression to depression.

```
27          lines[i] = lines[i].replace('#', '')
```

iv. Remove any URL.

```
30          lines[i] = re.sub(r"http\S+", "", lines[i])
31          lines[i] = re.sub(r"www\S+", "", lines[i])
32          lines[i] = lines[i].strip()
```

v. Convert every word to lowercase.

```
35          lines[i] = lines[i].lower()
```

vi. Removed all punctuations and other symbols.

```
38          lines[i] = re.sub('[^A-Za-z0-9 ]+', '', lines[i])
39          lines[i] = " ".join(lines[i].split())
```

3. Execution Instruction

```
python main.py main.py --file file_path --max_k max_k
optional arguments:
  --file path to the .txt file (default: data/gdnhealthcare.txt)
  --max_k maximum number of clusters to use (default: 10)
```

4. Results

Jaccard distance is used to calculate distance between two sentences:

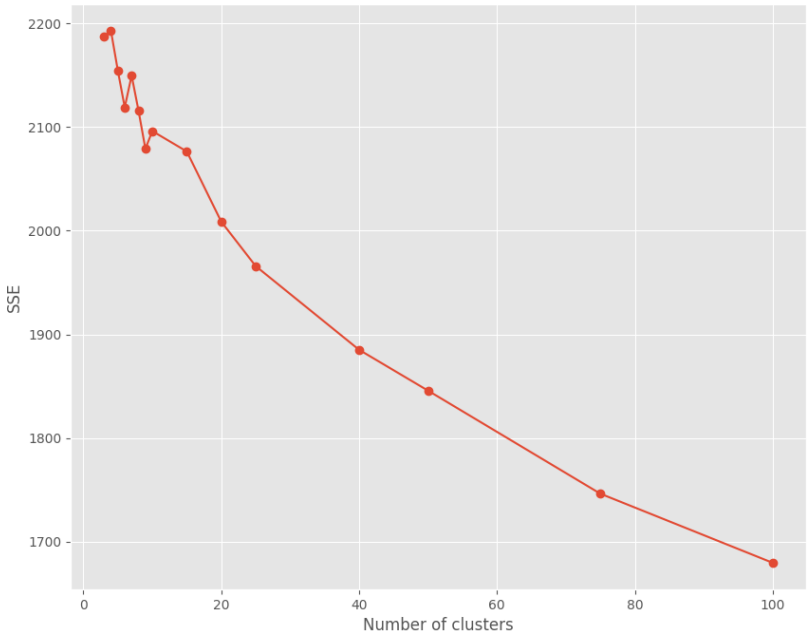
```
72    def jaccard_distance(t1, t2):
73            return 1 - (len(set(t1).intersection(t2)) /
74                        len(set().union(t1, t2)))
```

Elbow plot for gdnhealthcare.txt:

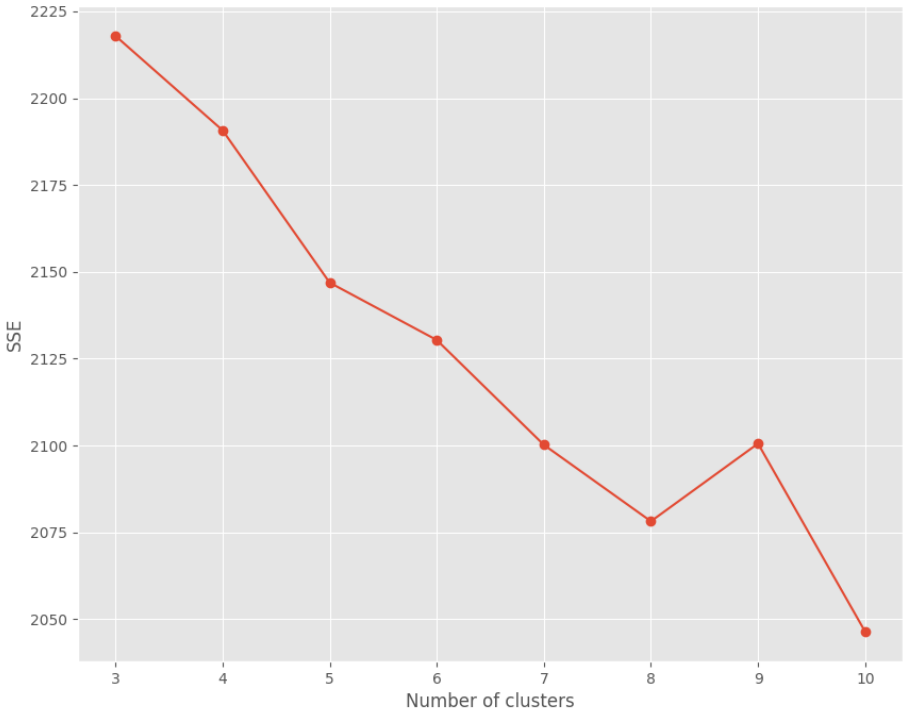
The plot seems to indicate a slight elbow at around 20-40 clusters. This suggests that increasing the number of clusters beyond this point may not significantly improve the clustering results, while adding more computational cost.

Data distribution in clusters:

The length of each cluster varies significantly across different k values. For instance, with k set at 3, Cluster 0 has a length of 1048, while Cluster 2 has a length of only 303. This suggests that the data points may not be uniformly distributed across the clusters.
It is important to note that k-means clustering is sensitive to the initial choice of centroids. Different initializations can lead to different clustering.
Improvement: As this assignment is using Jaccard distance, it does not take into consideration the importance, context or meaning of the words in tweets. Techniques like cosine similarity or more advanced techniques like text embeddings will lead to better results.



For experiments, we have set value of k from 3 to 10. The SSE and cluster distribution is noted below:



k	SSE	Clusters
3	2217.93	Cluster 0 Length: 1048 Cluster 1 Length: 1489 Cluster 2 Length: 303
4	2190.75	Cluster 0 Length: 1708 Cluster 1 Length: 594 Cluster 2 Length: 82 Cluster 3 Length: 456
5	2146.89	Cluster 0 Length: 136 Cluster 1 Length: 553 Cluster 2 Length: 300 Cluster 3 Length: 1271 Cluster 4 Length: 580
6	2130.37	Cluster 0 Length: 623 Cluster 1 Length: 474 Cluster 2 Length: 331 Cluster 3 Length: 271 Cluster 4 Length: 840 Cluster 5 Length: 301
7	2100.19	Cluster 0 Length: 76 Cluster 1 Length: 508 Cluster 2 Length: 367 Cluster 3 Length: 343 Cluster 4 Length: 692 Cluster 5 Length: 162 Cluster 6 Length: 692
8	2078.18	Cluster 0 Length: 504 Cluster 1 Length: 258 Cluster 2 Length: 363 Cluster 3 Length: 459 Cluster 4 Length: 596 Cluster 5 Length: 289 Cluster 6 Length: 53 Cluster 7 Length: 318
9	2100.56	Cluster 0 Length: 417 Cluster 1 Length: 355 Cluster 2 Length: 274 Cluster 3 Length: 527 Cluster 4 Length: 237 Cluster 5 Length: 802 Cluster 6 Length: 50 Cluster 7 Length: 93 Cluster 8 Length: 85
10	2046.42	Cluster 0 Length: 174 Cluster 1 Length: 283 Cluster 2 Length: 371 Cluster 3 Length: 666 Cluster 4 Length: 282 Cluster 5 Length: 378 Cluster 6 Length: 188 Cluster 7 Length: 88 Cluster 8 Length: 220 Cluster 9 Length: 190