

Solution code- [A3.3 Q1](#) [A3.3 Q2](#)

Q1

- a. **Load the file into appropriate data structures so that the subsequent operations can be done.**

Ans: I loaded the file in python. In 'Overall Rating' there are some values such as 'New' and '-'. The dataset is cleaned and now it is (658x7). Also, the numerical values are converted into float datatypes.

- b. **Generate a new file "cafe_sentiment.csv" which contains only the reviews and a label - where the labels are generated as follows: if (rating > 3.5), then "GOOD" else "BAD"**

Ans: The Overall Rating values are converted into Labels (Good/Bad) and a new column is created. This file comprising of two columns 'Review' and 'Label' is saved as the new .csv file.

- c. **Develop a text classifier using any model using the SciKit Learn library. Use 70 to 80% of data as training and remaining as test data.**

Ans: First step is tokenising text using CountVectorizer() to build a dictionary of features and transform documents to feature vectors. This gives a matrix of 658x2959. Using TfidfTransformer (Term Frequency times Inverse Document Frequency) function the estimator is fit to the data and then count matrix is transformed into tf-idf representation. The following models are built on the train and test

1. Naive Bayes Classifier:
2. SGD Classifier:

- d. **What is the performance of your model? Which classifier did you choose and why?**

Q2 Named Entity Recognition with Dataset

- a. **Write code to perform Named Entity Recognition using Spacy and SciSpacy libraries to detect the Drugs from the statements.**

Ans: Named Entity Recognition identifies and extracts essential information from text. NER is a NLP method that detects and categorizes named entities in text, including people, organizations, locations, dates, quantities, and other identifiable real-world entities.

- b. **Compile the output of NERs for each statement and generate an aggregate table as follows:**

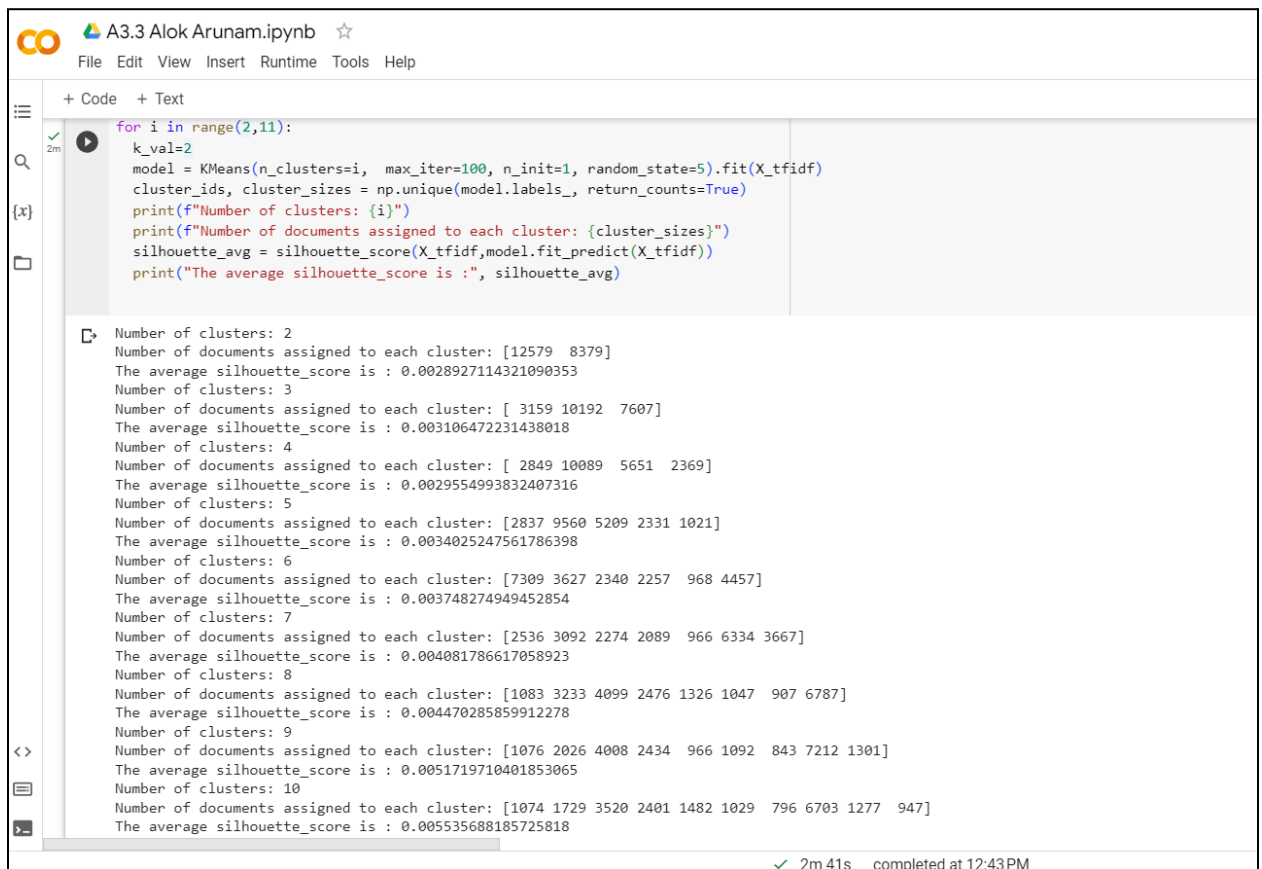
For each unique output of the NERs - find the total of number times it occurs in the dataset, number of times it occurs in statements with POS label and number of times it occurs in statements with NEG label.

Q3) Write code to run k-means clustering algorithm over the bio-medical statements in the above-mentioned file.

a. Starting with k=2, you can go up to k=10 or more.

Ans: The first step is to tokenise text and transform using `TfidfVectorizer()` to build a dictionary of feature vectors. Using 'fit_transform' this is fit to the data and then count matrix is transformed into tf-idf representation. This gives a matrix of 20958x17122.

The next step is to import `KMeans` and build model.



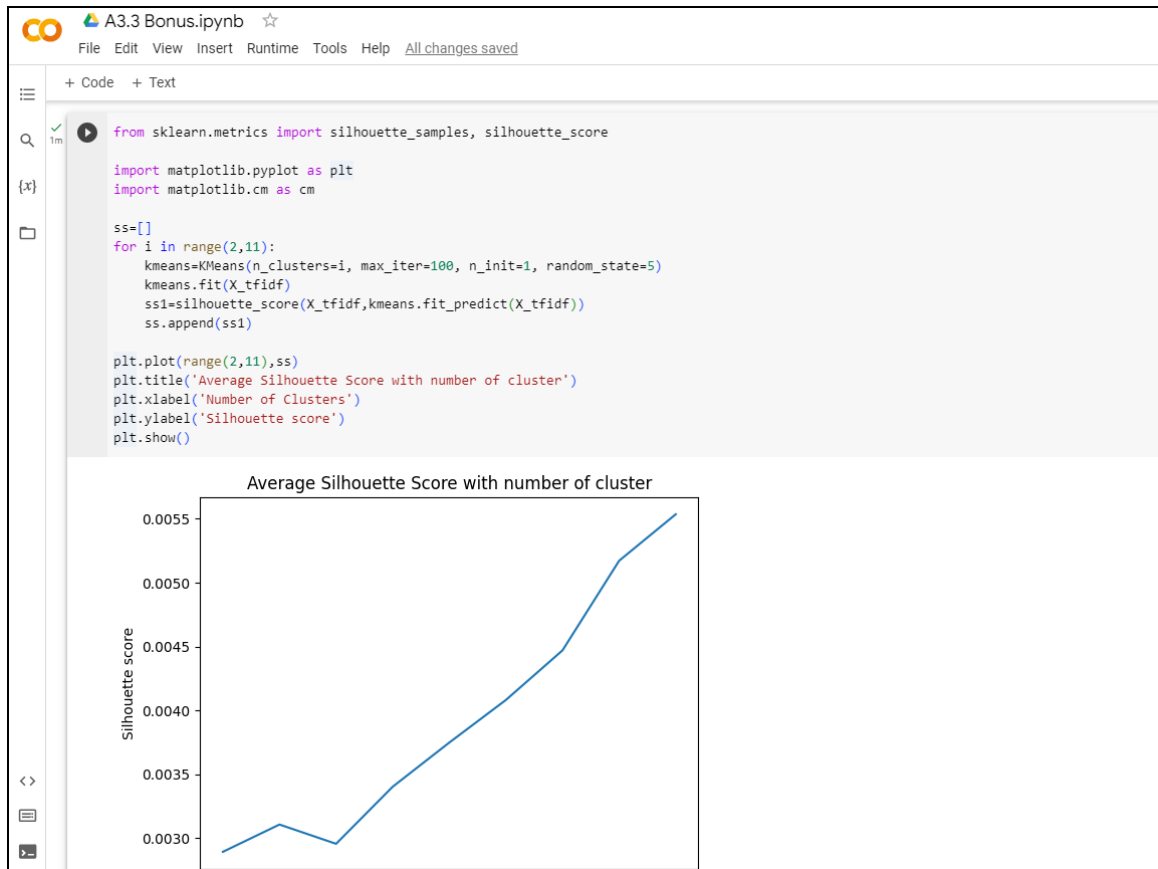
```
for i in range(2,11):
    k_val=i
    model = KMeans(n_clusters=i, max_iter=100, n_init=1, random_state=5).fit(X_tfidf)
    cluster_ids, cluster_sizes = np.unique(model.labels_, return_counts=True)
    print(f"Number of clusters: {i}")
    print(f"Number of documents assigned to each cluster: {cluster_sizes}")
    silhouette_avg = silhouette_score(X_tfidf,model.fit_predict(X_tfidf))
    print("The average silhouette_score is :", silhouette_avg)
```

Number of clusters: 2
Number of documents assigned to each cluster: [12579 8379]
The average silhouette_score is : 0.0028927114321090353
Number of clusters: 3
Number of documents assigned to each cluster: [3159 10192 7607]
The average silhouette_score is : 0.003106472231438018
Number of clusters: 4
Number of documents assigned to each cluster: [2849 10089 5651 2369]
The average silhouette_score is : 0.0029554993832407316
Number of clusters: 5
Number of documents assigned to each cluster: [2837 9560 5209 2331 1021]
The average silhouette_score is : 0.0034025247561786398
Number of clusters: 6
Number of documents assigned to each cluster: [7309 3627 2340 2257 968 4457]
The average silhouette_score is : 0.003748274949452854
Number of clusters: 7
Number of documents assigned to each cluster: [2536 3092 2274 2089 966 6334 3667]
The average silhouette_score is : 0.004081786617058923
Number of clusters: 8
Number of documents assigned to each cluster: [1083 3233 4099 2476 1326 1047 907 6787]
The average silhouette_score is : 0.004470285859912278
Number of clusters: 9
Number of documents assigned to each cluster: [1076 2026 4008 2434 966 1092 843 7212 1301]
The average silhouette_score is : 0.0051719710401853065
Number of clusters: 10
Number of documents assigned to each cluster: [1074 1729 3520 2401 1482 1029 796 6703 1277 947]
The average silhouette_score is : 0.005535688185725818

2m 41s completed at 12:43PM

b. Compute the silhouette score for each run and plot a graph to show how they change with k.

Ans: Silhouette score increases with k as shown in the plot.



- c. Write a program that can determine automatically the value for “k” that finds the best segregation of POS and NEG statements. (Hint: k may be greater than 2. Use Purity measures.)