REPORT
Assignment 2

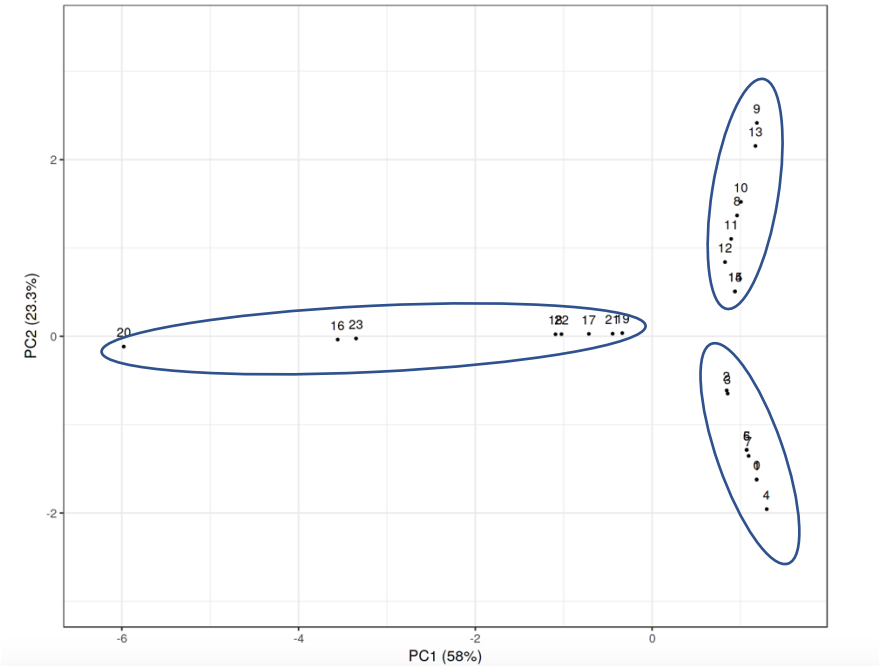
Inavamsi Enaganti, <ibe214@nyu.edu>

Contents

Aim: Clustering a given set of text documents

For the visualization, I took the threshold frequency to be 35+ to filter away most words. Finally there were the following topics left over. Applying PCA gave me the following result. As seen in the file matrix_visualise.csv file with high enough threshold frequency. Removed word 'yen' as it appears in only 1 file.

airline rate disease loan bank safety mortgage



PC1 Variance = 51.7% PC2 Variance = 27.3%

Output
Output files:
topics.txt - topics in text format
matrix_visualise.csv - matrix used to visualize in csv format
lem_tok.txt - tokenized and lemmatized words in text format
NER.txt – NER terms in text format

Sample Output:
Matrix output to file: matrix_visualise.csv

Distance Metric selected: cosine
Total clusters selected: 3

Kmeans++
cluster0 : C1/article01 C1/article02 C1/article03 C1/article04 C1/article05 C1/article06 C1/article07 C1/article08
cluster1 : C4/article01 C4/article02 C4/article03 C4/article04 C4/article05 C4/article06 C4/article07 C4/article08

cluster2 : C7/article01 C7/article02 C7/article03 C7/article04 C7/article05 C7/article06 C7/article07 C7/article08

Kmeans
cluster0 : C4/article07 C7/article02 C7/article05 C7/article06
cluster1 : C1/article01 C1/article02 C1/article03 C1/article04 C1/article05 C1/article06 C1/article07 C1/article08
cluster2 : C4/article01 C4/article02 C4/article03 C4/article04 C4/article05 C4/article06 C4/article08 C7/article01 C7/article03 C7/article04 C7/article07 C7/article08

Confusion Matrix

For Kmeans++ with cosine similarity
cluster0 : C1/article01 C1/article02 C1/article03 C1/article04 C1/article05 C1/article06 C1/article07 C1/article08
cluster1 : C4/article01 C4/article02 C4/article03 C4/article04 C4/article05 C4/article06 C4/article07 C4/article08
cluster2 : C7/article01 C7/article02 C7/article03 C7/article04 C7/article05 C7/article06 C7/article07 C7/article08

N=24	Predicted Cluster 0	Predicted Cluster 1	Predicted Cluster 2	
Actually Cluster 0	8	0	0	TC0= 8
Actually Cluster 1	0	8	0	TC1= 8
Actually Cluster 2	0	0	8	TC2= 8
	TP0= 8	TP1= 8	TP2= 8	

Where Cluster 0 is C1, Cluster 1 is C4, Cluster 2 is C7.
Recall =TPi/(TPi+FNi) = TruePositive(i)/Actual size of cluster i
Precision =TPi/(TPi+FPi) = TruePositive(i)/Total predicted as i
Recall 0 = 8/8 =1 Precision 0 = 8/8 =1
Recall 1 =8/8 = 1 Precision 1=8/8 =1
Recall 2 =8/8 = 1 Precision 2=8/8 =1

Average Recall = 1 Average Precision =1
F-measure = 2PR/(P+R) =1

For Kmeans with cosine similarity
cluster0 : C1/article01 C1/article02 C1/article03 C1/article04 C1/article05 C1/article07 C7/article01 C7/article02 C7/article03 C7/article07 C7/article08
cluster1 : C1/article06 C1/article08 C7/article04 C7/article05 C7/article06
cluster2 : C4/article01 C4/article02 C4/article03 C4/article04 C4/article05 C4/article06 C4/article07 C4/article08

N=24	Predicted Cluster 0	Predicted Cluster 1	Predicted Cluster 2	
Actually Cluster 0	6	2	0	TC0= 8
Actually Cluster 1	5	3	0	TC1= 8
Actually Cluster 2	0	0	8	TC2= 8
	TP0= 11	TP1= 5	TP2= 8	

Recall =TPi/(TPi+FNi) = TruePositive(i)/Actual size of cluster i
Precision =TPi/(TPi+FPi) = TruePositive(i)/Total predicted as i
Recall 0 = 6/8 =0.75 Precision 0 = 6/11= 0.55
Recall 1 =3/8 = 0.37 Precision 1=3/5 =0.6
Recall 2 =8/8 = 1 Precision 2=8/8 =1

Average Recall = 0.71 Average Precision = 0.72
F-measure = 2PR/(P+R) = 0.715

Readme

Tools used:

- Stanford CoreNLP - to install visit https://stanfordnlp.github.io/CoreNLP/index.html#download
- Eclipse - IDE for Java code
- To use Stanford CoreNLP on Eclipse using Maven- http://www.sfs.uni-tuebingen.de/~keberle/NLPTools/presentations/CoreNLP/NLP%20Tools%20-%20Stanford%20CoreNLP%20-%20Installation%20(1).pdf
- ClustVis - Visualise Data - https://biit.cs.ut.ee/clustvis/

Running the Project in Eclipse IDE:

Load App.java and run it(it contains the main method). Located at PA2/src/main/java/PA2NLP/PA2/App.java

Required files from folder: readFile.java, Preprocess.java, Merge.Java, Matrix.java, kmeans.java, kmeanspp.java, csv.java, pom.xml

Dependency files:

```
<dependencies>
  <dependency>
    <groupId>junit</groupId>
    <artifactId>junit</artifactId>
    <version>3.8.1</version>
    <scope>test</scope>
  </dependency>
  <dependency>
        <groupId> edu.stanford.nlp </groupId> <artifactId> stanford-corenlp </artifactId>
<version>3.8.0</version>
  </dependency>
  <dependency>
        <groupId> edu.stanford.nlp </groupId> <artifactId> stanford-corenlp </artifactId>
<version>3.8.0</version>
        <classifier> models-english </classifier>
  </dependency>
  <dependency>
  <groupId> org.apache.cassandra</groupId>
  <artifactId>cassandra-all</artifactId>
  <version>0.8.1</version>
      </dependency>
      <!-- https://mvnrepository.com/artifact/ch.qos.logback/logback-classic -->
      <dependency>
        <groupId>ch.qos.logback</groupId>
        <artifactId>logback-classic</artifactId>
        <version>1.0.13</version>
        <scope>test</scope>
      </dependency>
```

PS: don't forget the log4j.properties file

Supplying New Parameters

Adding new files:

In class main in App.java there is a string called filenames, add the name of the text file without the txt. Add the required file in the data folder PA2/data.

Example: To add test.txt, add "test" to string fileNames and test.txt to the data folder.

Changing parameter values

Number of clusters: Change the variable clusters in line 18 in App.java. The variable is in class main. Default value set to 3.

Distance Metric: Change the variable choice in line 20 in App.java. The variable is in class main. Default value set to 1. 1-cosine similarity, 2 - euclidean metric

Threshold frequency: default value set to number of documents.