Text Mining

Assignment: 3

(15 points)

Q.1 (0 Point)
NO RESPONSE REQUIRED
Verify the results of Table 4.1 (see text). You can write a small program based on Figure 4.5 (see text) or use an EXCEL spreadsheet to do this. No response is required.

Q.2 (2 Points)
Take any business article from your favorite online newspaper, and create an xml file by tagging it the same way as the Reuters articles. You can do this as simply as cut/paste of article text and adding the relevant tags. You can unpack the Reuters corpus into a temporary directory and see the individual articles to see examples of what you must create.

Now use this file to retrieve the 10 best matches from the Reuters training XML. To do this, create query and document vectors first by using 'create_document_vectors' and 'create_query_vectors' functions in 'ret' module. Note you need to create a properties file for query document to do this task. For finding the 10 most similar documents, use 'find_similar_k_docs' function in the same module. Comment on the quality of the matched results.

Or if you wish to use TMSK

Now use this file to retrieve the 10 best matches from the Reuters training XML documents using the TMSK routine "matcher". Comment on the quality of the matched results. Note that you must have created the inverted index file for the training vectors (see the TMSK documentation). What are the key determining factors for the quality of the matched results?

Q.3 (10 Points)
Use the k-means program to generate 10 clusters of the Reuters training data. Re-run to generate 20 clusters, then 30 clusters. Discuss the sorts of clusters you get and any hints they might provide as to the optimal k for this document collection. Use 'cluster_k_means' module to do this task. Report the output of clustering algorithm by printing the silhouette coefficient score obtained from the above module.

Or if you wish to use TMSK

Use the k-means program to generate 10 clusters of the Reuters training data. Re-run to generate 20 clusters, then 30 clusters. Discuss the sorts of clusters you get and any hints they might provide as to the optimal k for this document collection. Note that the program reports the mean cosine distance per case: it measures the cosine distance of each case from the mean of the bin to which it is assigned, then takes the mean (of all these distances). See Equation 4.3 of the book for cosine distance. This is the preferred distance measure in world of text data. One can also compute the total variance from the mean as shown on Equation 5.1. Create a text file with a log of your cluster descriptions (as generated by the program) and also your discussion. Upload this file as your answer to the question.

Q.4 (0 Point)
NO RESPONSE REQUIRED
Verify the computations in Figure 5.13 (see text) using the algorithm in Figure 5.12 (see text). No Response is required.

Q.5 (2 Points)
How can the EM algorithm be used to perform a k-means clustering?

Q.6 (1 Point)
List any one advantage of using the EM algorithm over the k-means algorithm.

<u>Optional</u> part for students with knowledge of Python

Questions 2 and 3 can be done in Python instead of using TMSK. Since they build on the Python solution of assignment-1, be sure to try the model solution of that assignment and make sure you run it successfully and obtained the data needed for this assignment.

We have provided a sample python module 'ret' along with the python modules 'cls' and 'tm' from previous assignments that contain some basic building blocks useful for this assignment. You can freely import 'ret', 'cls' and 'tm' modules to write your answers.