CS 280 Course Project Stage 2

Due: Before class Monday Jan 6 (report & presentation)

Project 2 will continue on project 1. In project 2, you will create vector representations of training and test documents with terms identified in project 1, and learn and test a classifier.

1. Outcomes

1.1 Particularly, you need to create two files, one file for vector representation of all documents in the training set, and one file for vector representation of all documents in the test set. The vector representation should be using TF-IDF values, and the format of each file should be as follows (you do not need to actually store headers w1, w2 ... and d1, d2, ... in the files but corresponding matrix tf-idf values):

	w1	w2	w3	.....	w1000
d1					
d2					
.					
.					
.					

Here, w1, w2, w3 ... are top 200 most frequent terms you identified from the dataset combining both the training and test sets in project 1. This 200 terms will be used as the vocabulary to represent each document in training set and test set. The vector tf-idf representation for the training set should be stored in one file, and the vector tf-idf representation for the test set should be stored in another file.

1.2 Using matrix data obtained from the training set, calculate centroids of 3 types of documents. Since the training data set contains 3 types of documents and you have converted these documents into vector representation, calculate the means of each type of documents as the centroid of the type of document. The 3 controids will be used as a classifier model.

1.3 Test your classifier on the test data set obtained from 1.2 and give the accuracy: how many test documents your classifier can correctly classify. You may use Euclidean distance or cosine similarity to do classification.

2. TF-IDF by data analysis

You may write your own code to analyze the documents to get tf-idf values for each document vector or research and use any available library packages from *R, Weka, Python, or other ML* packages.

3. Report

Report should be formally organized, formatted, and written without spelling errors. Report should clearly describe the problem you address, and particularly how you address the problem step by step with code details. Provide references at the end of report for any work you refer.

4. Presentation

Presentation should include code, vector results demonstration, and accuracy results, and problems encountered and solutions if there are.