Cmpe 493 Introduction to Information Retrieval, Fall 2015
Assignment 2 - Spam Email Filtering, Due: 10/12/2015 (Thursday), 23:00

In this assignment you will implement a spam/non-spam filter using the Rocchio and the k-Nearest Neighbor (kNN) algorithms. You will use a subset of the Ling-Spam corpus[1] to train and test your system. The provided training and the test sets (included in the *dataset.zip file*) each contain 240 spam and 240 legitimate email messages. Each email message is provided as a separate file. All files start with a "subject:" heading. Stopword removal and lemmatization have already been performed.

Preprocess the files by extracting the individual tokens from them. You can implement your own tokeniser or use a tokeniser from an available library or tool. Represent each document (email message) as a normalised TF-IDF weighted vector. The dimensionality of your vectors should be equal to the size of your vocabulary, which you should learn from the training set only.

kNN is a simple, non-parametric and widely used classification algorithm. It classifies a new document according to the majority vote of its k-Nearest Neighbors. Given a test document, you should compute the k most similar documents in the training set and assign it to the class (spam or non-spam) of the majority of these neighbors. For example, if k = 1, then a new unclassified document (test document) will be classified with the same class as its nearest neighbor (i.e., the most similar document) in the training set. You should use cosine similarity as your similarity function both for the kNN and the Rocchio algorithms.

Submission: You should submit a *".zip"* file named as YourNameSurname.zip containing the following files using the Moodle system:

1. Report:
   (a) Briefly describe how you performed tokenisation.
   (b) What is the size of your vocabulary?
   (c) For each word in the vocabulary, compute the sum of the TF-IDF scores of the word in all document in the spam training set. List the top 20 words with the highest total TF-IDF scores in the spam training set.
   (d) For each word in the vocabulary, compute the sum of the TF-IDF scores of the word in all document in the non-spam (legitimate) training set. List the top 20 words with the highest total TF-IDF scores in the non-spam training set.
   (e) Report the precision, recall, and F-measure values of your system for identifying spam email messages. The results for the Rocchio algorithm as well as for kNN with k = 1,3,5,7,9 should be reported in a table.
   (f) Compare the results obtained by the kNN and the Rocchio algorithms in a paragraph. State which algorithm performs better than the other. Justify your claim.

[1] I. Androutsopoulos, J. Koutsias, K.V. Chandrinos, George Paliouras, and C.D. Spyropoulos, "An Evaluation of Naive Bayesian Anti-Spam Filtering". In Potamias, G., Moustakis, V. and van Someren, M. (Eds.), Proceedings of the Workshop on Machine Learning in the New Information Age, 11th European Conference on Machine Learning (ECML 2000), Barcelona, Spain, pp. 9-17, 2000.

2. Commented source code, executable, and readme: You may use any programming language of your choice. However, I need to be able to test your code. Submit a readme file containing the instructions for how to run your code.

Late Submission: You are allowed a total of 5 late days on homeworks with no late penalties applied. You can use these 5 days as you wish. For example, you can submit the first homework 2 days late, then the second homework 3 days late. In that case you will have to submit the remaining homeworks on time. After using these 5 extra days, 10 points will be deducted for each late day.