CmpE 493
Information Retrieval

Assignment 2 Report
Atakan Arıkan – 2011400243

(a) Briefly describe how you performed tokenisation.

It's a simple tokenizer, splits the given line by whitespace. Then for each token, if the token is not convertible to the double, iterates through each character and gets rid of the nonword characters. For instance, U.S.A. becomes USA.

(b) What is the size of your vocabulary?

> 14832

(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.

> 0, money, mail, address, free, business, internet, http, day, 20, remove, com, 100, order, check, site, list, product, report, credit

(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 training set. List the top 20 words with the highest total TF-IDF scores in the non-spam training set.

> language, university, linguistic, linguist, linguistics, english, department, edu, study, reference, issue, theory, speak, word, student, de, grammar, seem, science, speaker

(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.

Rocchio:

TP: 240,

TN: 240,

FP: 0,

FN: 0

Precision: 1,

Recall: 1,

F-measure: 1

1-NN:

TP: 240,

TN: 240,

FP: 0,

FN: 0

Precision: 1,

Recall: 1,

F-measure: 1

3-NN:

TP: 240,

TN: 240,

FP: 0,

FN: 0

Precision: 1,

Recall: 1,

F-measure: 1

5-NN:

TP: 240,

TN: 235,

FP: 5,

FN: 0

Precision: 0.979,

Recall: 1,

F-measure: 0.989

7-NN:

TP: 240,

TN: 235,

FP: 5,

FN: 0

Precision: 0.979,

Recall: 1,

F-measure: 0.989

9-NN:

TP: 240,

TN: 107,

FP: 133,

FN: 0

Precision: 0.979,

Recall: 1,

F-measure: 0.989133

(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.

Rocchio is much faster than kNN algorithm since it only considers the centroid. It doesn't calculate the cosine similarity for all the documents in a set. And in my code, it gave the perfect result for spam/legitimate cases.

KNN is more accurate on the paper since it considers each document seperately. However, it relies heavily on the processing power and takes a long time to output results. Also, if we select a small number for "k", we'd be wasting lots of documents. By wasting I mean we calculate the cosine similarity, but we do not consider the result when we're making a decision.