CS 510
Information Retrieval on the Internet
Spring, 2010

Written Assignemnt 3: Text Classification and Support Vector Machines

Due Tuesday, May 4

Part A: Written Problems

1. Consider the following training set of two-dimensional vectors.

Positive class: (-2, 2), (-1, 1), (0, 1) Negative class: (0, -1), (1, 1), (2, 2)

Suppose you are given a new vector, (1/2, 1/2).

(a) Show how it would be classified under Rocchio classification.

(b) Show how it would be classified under 1-nearest neighbor classification

(Show your calculations for parts (a) and (b).)

2. Show that Rocchio classification can assign a label to a document that is different from its training set label. (Textbook Exercise 14.2)

3. Explain why kNN handles multimodal classes better than Rocchio. (Textbook Exercise 14.2)

4. Consider the confusion matrix given in table 14.5 of the textbook. What is the precision and recall of each of the six classes? What is the average precision and recall of the classifier?

Part B: Experimenting with a Support Vector Machine

In this part of the assignment you will experiment with using a support vector machine on the UCI spam classification data set.

The support vector machine system you will use is called SVM-Light. It was written by Thorston Joachims. You can download the code from http://svmlight.joachims.org/ This web site also gives instructions on how to compile and run the code.

Here is what you need to do for this assignment:

1. Download the SVM_light code from the URL given above, and read the instructions on how to run it. You can simply use all the default parameters, which means you don't need to specify any options (except for the experiments below on varying the kernel type.)

2. Download the training ("spam.svm.train") and test ("spam.svm.test") files from the class web page. The training file now has 960 instances and the test file has 3681 instances, each with about 39.5% spam and 59.5% non-spam.

3. Run svm_learn on the training data (this will take a few minutes). Then run svm_classify on the test data. Report the following (you can get all this information from the output of the SVM code and from the "model" file it creates):

- The classification accuracy (i.e., fraction of correct classifications) on the training data.
- The classification accuracy (i.e., fraction of correct classifications) on the test data.
- The precision and the recall on the test data.
- The number of support vectors that were used in the model created by the SVM.

4. In step 3 you used the default *linear* kernel. Repeat step 3 using two other kernels (specified with the -t option): *polynomial* (1) and *radial basis function* (2). Report the same results for each of these that you reported in step 3.

5. Briefly answer the following questions:

- Which kernel gives you the best generalization performance?
- Does the number of support vectors seem related to the generalization performance on different kernels?
- Do you see any evidence of over-fitting? Explain.
- Why do you think the precision is often (perhaps always) larger than the recall in the results of SVMs on this problem?
- Why do you think the classification accuracy on the test data sometimes higher than on the training data?