CS3244 Machine Learning
Assignment 2

Shawn Tan (U096883L)

1 Introduction

Text classification is a common problem in the field of machine learning and Natural Language Processing (NLP). In this assignment, we were tasked to classify some posts on several newsgroups.

We were given stemmed texts from 5 newsgroups: `comp.graphics`, `comp.os.ms-windows.misc`, `comp.sys.ibm.pc.hardware`, `comp.sys.mac.hardware` and `comp.windows.x`. In our test set, we were given 1425 instances to classify, and 2935 training instances. Several scripts and programs were supplied to perform various tasks:

fs.php and fe.php These two PHP scripts help to extract the features from the texts using TF-IDF and χ^2 feature selection methods.

Formatting.exe This program converts the `.txt` files created by the PHP scripts into `.arff` files which can be read by WEKA.

The end result are two `.arff` files that consist of features that correspond to normalised word frequencies. These are the feature vectors which the various classifiers used will be working with. In this report, we experiment with using 3 different types of classifcation algorithms: k-Nearest Neighbour, Naive Bayes, and SVMs.

Our approach involves training different classifiers using each of the algorithms using the same dataset. Eventually, we take the best performing classifier from each different algorithm, and use these classifiers together to hopefully reduce any kind of overfitting caused by any of the individual algorithms. After evaluating this classifier, we then use this to classify our test set.

We make use of version 3.7.4 of WEKA for the tasks detailed in this report.

2 Selecting the Features

Setting an overly high value for feature selection may result in feature vectors that are too specific to the training set, and eventually cause overfitting. For our first experiment, we select only the top 50 keywords for each class for our feature vector. This resulted in 203 keywords in total.

Using the selected features, we extract the feature vectors from each of the newsgroup posts. Using this, we train three classifiers (kNN, Naive Bayes, SVM) using the default WEKA settings, and evaluate their performances before proceeding. We do this several times, with

fs_top_num	Keywords/Features	**Naive Bayes**	**SVM**	**IBk**
50	203	0.738	0.77	0.732
100	428	0.74	0.801	0.758
150	641	0.743	0.814	0.767
200	857	0.74	0.822	0.774

Table 1: Experiments with the number of features used.

several different values of `fs_top_num`. Table 2 reports the different values we tried, and the F-Measure of the corresponding classifiers.

Increasing `fs_top_num` by 50 at each round of testing, we performed the experiment four times. We decided to use an `fs_top_num` value of 200 for our classifcation task, as larger feature vectors may cause classification to take long periods of time, making repeated testing difficult. In the following section we attempt to tune the performance of individual classifiers by adjusting the parameters for the different algorithms.

3 Tuning Performance of Individual Classifiers