# Lab 3: Classification

# Introduction

The purpose of this lab was to perform different classification methods on the Reuters news articles dataset seen in previous labs. For this lab I chose to test three different classifiers: K Nearest Neighbors, Naïve Bayesian, and Decision Tree. The performance of these classifiers was evaluated in terms of scalability and accuracy.

# Preprocessing

Since the last lab, there was only one change in preprocessing. I used the python library sci-kit learn to implement the three classifiers, and not all of these classifiers support use of multi-class labels. In order to standardize comparisons of classification algorithms therefore, I condensed all class labels which had multiple topics into just the topic that was most prevalent in the overall dataset. For example, as ‘earn’ was a very common topic, and ‘sun-oil’ was comparatively rare, a class label of [‘earn’, ‘sun-oil’] would be condensed to only ‘earn’. Additionally, I only used the feature vector containing both topic keywords and body keywords, as in the clustering lab it gave better results in all testing.

# Classification

As mentioned before, all three algorithms used were from the sklearn package. For each of the three classification methods, I tested different training splits of the data. The three splits analyzed were 50% training and 50% testing, 66% training and 33% testing, and 80% training and 20% testing. For each of these split ratios the following four measures were recorded:

Offline Training Time

The time in milliseconds it took to fit the classifier to one tuple of data. The total time for the fitting was divided by the number of tuples in the training set.

Online Training Time

The time in milliseconds it took the classifier predict the label of one tuple of data. The total time for predicting all of the testing labels was divided by the total number of tuples in the testing set.

Precision

This was the result of an sklearn function which returns the weighted ratio of true positive classifications to the sum of the true and false positive classifications.

Accuracy

This is the percent of samples correctly clustered. Since the original ground truth labels were condensed into single topics, the overall accuracy for each classification method was at the low end of the spectrum for what may actually be true. If any overlap in predicted versus actual labels were counted as correct (i.e. [‘earn’, ‘acq’] predicted with [‘earn’, ‘money-fx’, ‘grain’] actual is reported as a positive classification) the accuracy measure of, for example, KNN went up by 10% on average.

# Results

The results for K Nearest Neighbors, Naïve Bayesian, and Decision Tree algorithms can be seen and compared in figure 1, 2, and 3 respectively below and on the following page.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| K Nearest Neighbors Classifier | | | | |
| Training Split | Offline Time per tuple (ms) | Online Time per tuple (ms) | Precision | Accuracy |
| 50/50 | 0.082 | 8.272 | 0.6796 | 0.6906 |
| 66/33 | 0.096 | 9.963 | 0.6941 | 0.7123 |
| 80/20 | 0.115 | 14.245 | 0.7025 | 0.6775 |

**Figure 1**: KNN classifier results on full Reuters dataset.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Naïve Bayesian Classifier | | | | |
| Training Split | Offline Time per tuple (ms) | Online Time per tuple (ms) | Precision | Accuracy |
| 50/50 | 0.0171 | 0.6346 | 0.6425 | 0.6087 |
| 66/33 | 0.0169 | 0.6918 | 0.6612 | 0.6145 |
| 80/20 | 0.0169 | 0.7385 | 0.6473 | 0.5999 |

**Figure 2**: Naïve Bayesian classifier results on full Reuters dataset.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Decision Tree Classifier | | | | |
| Training Split | Offline Time per tuple (ms) | Online Time per tuple (ms) | Precision | Accuracy |
| 50/50 | 0.1917 | 0.0030 | 0.6920 | 0.6979 |
| 66/33 | 0.2062 | 0.0031 | 0.6966 | 0.7041 |
| 80/20 | 0.2297 | 0.0031 | 0.6834 | 0.6891 |

**Figure 1**: Decision Tree classifier results on full Reuters dataset.

Scalability

In terms of offline time for training a classifier, the Naïve Bayesian model was the quickest, with a very scalable time of only about 0.017ms per tuple. This equated to a total time for the entire Reuters dataset (trimmed to around 10500 samples with topics) of only 0.18 seconds. The worst classifier for offline training time was the Decision Tree, which still only took about 3 seconds to train on the majority of the dataset. For online time, Decision Tree came in orders of magnitude faster than the others, and given 1 million samples of data to classify, it would only take 3.1 seconds to finish compared to KNN taking upwards of 3-4 hours.

Training split

For each of the three algorithms, the training split seemed to have small effect on the accuracy of the classification. There was the consistent trend of 66/33 being the best split as far as accuracy was concerned, where going above or below that would decrease the accuracy slightly. Accuracy and Precision

Precision and accuracy measures were almost the same for both K Nearest Neighbors and Decision Tree classifiers in every split. For Naïve Bayesian the split was slightly more significant, with the precision being higher than accuracy by about 4% in every case. Similarly, K nearest Neighbors and Decision Tree classifiers produced the best accuracy, with around 71% each, while Bayes only correctly classified about 61% correctly.

# Conclusion

From the results above, there are a couple of key takeaways. The most scalable algorithm was the Decision Tree classifier, which once fit to data can predict labels for data in very little time due to its logarithmic time complexity based tree structure. The most accurate classifier, was the K Nearest Neighbors, which produced a maximum accuracy of 71.23%. This, however, was the least scalable of the three algorithms tested, and also only marginally outperformed the Decision Tree classifier. Therefore, for this dataset, and other like it, I would recommend Decision Tree classification as the overall best of the three algorithms.