IR program HW2 R02922083 邵元輔

The Naïve Bayes Classifier is implemented in nbc.py. There are several techniques were used to improve performance. I lowered each word and separated with stemming to minimize the concerned words. And then build the vocabulary with a tunable stop frequency. That is omit the word with too often happen in document. The Naïve bayes classifier is calculate through 2 function NBScore() and computeNaiveBayesClass(). The first function is to create the probability estimate of each word in each category. With this estimates, we can then calculate the corresponding naïve bayes class of each test document through the formula in KAMAL NIGAM's paper. To tuning the performance, I also added tunable parameters in the nominator and denominator of probability estimate. In the end, use log10 to compare the max likelihood of each class to decide the class of each document. The result shows an optimal 0.7 accuracy after parameters tuning. The EM model is also followed paper's procedure. First, the E step use current classifier to calculate the class of each unlabeled data. And then in the M step combine the labeled and unlabeled data(with a label created from the last step) to create a new classifier. And literately use the EM step to refine the label of unlabeled data and the classifier to create a more accurate classifier. According to the paper, we also apply a small lambda to mitigate the problem of small label data. That is used lambda to tuning the weighting of unlabeled data. The result shows that for small label set. We can improve accuracy through EM process with a fine tuned lambda. The labeled data is highly correlated to the accuracy. For example, if we set label to 5 the accuracy is dropped to 0.2 and label 20 with accuracy about 0.5. However, with a full scale labeled data, the EM process could not improve the performance of prediction. The result is also mentioned in the experiment of the paper. Besides, each technique has a performance improvement in this system. For example, a system without stop word would dramatically dorp 0.5.

All labed data used:

With stemming+lemmantize =	0.706	(freq<1100)
Without stemming+lemmantize =	0.69	(freq<1100)
Without stemming+lemmantize =	0.68	(freq<1000)

Patial data used:

5 data label = 0.153

5 data label + unlabel 5 round with lamda 0.2 = 0.143

5 data label + unlabel 5 round with lamda 0.02 = 0.154

5 data label + unlabel 5 round with lamda 0.002 = 0.183

5 data label + unlabel 1 round with lamda 0.002 = 0.167

5 data label + unlabel 1 round with lamda 0.004 = 0.170

5 data label + unlabel 20 round with lamda 0.004 = 0.141

5 data label + unlabel 5 round with lamda 0.0002 = 0.166