CS4980 NLP Homework 3 Report

Grant Fass

The goal of homework 3 was to learn more about sentiment analysis by implementing a simple naive bayes model classifier. This classifier was to be trained and tested on a very small dataset to verify it worked properly. The next step was to try to add the MPQA subjectivity cues lexicon to the model and determine if that increased performance. The final step was to try to run the model on the Amazon Reviews dataset.

The original version of the classifier was somewhat inaccurate. It had an accuracy of 0.625. The precision and the recall were both 0.750. The F1-Score was 0.648. These values were retrieved from the original version of the model, before modifying the training set.

I then began to evaluate one of the documents the model classified incoreectly. The document I chose was `NEU It was a little slow, but not too bad.`. This document was supposed to be classified as Neutral. My classifier incorrectly classified it as Negative though. In order to determine why it was being mis-classified I performed a manual breakdown of each word in the document. I used the number of occurences of each word per class to calculate the sentiment probabilities for each word. I then computed the sum of the logs to get the overall value per class. The Negative class had a sum of -9.92. The Neutral class had a sum of -10.54. The Positive class had a sum of -10.91. The Negative class was the largest which is why it was predicted. I then looked at which terms were likely to be causing the misclassification. I determined that the words 'was' and 'slow' were probably the ones causing issues. I then added the document `NEU It was not too fast, but not too slow either.` to the training set. After retraining the model I found that the classifier had predicted the document correctly now. Additionally, the model accuracy improved to 0.750. The recall stayed the same at 0.750. The precision increased to 0.792. The F1-score increased to 0.750.

Next, I attempted to add the MPQA Subjectivity Cues Lexicon to the system. I first loaded the data into a pandas dataframe. I then remapped the 'both' class to 'neutral' and the 'weakneg' to 'negative'. I then added these words and their classes into the training data and retrained the model. The results of this ended up varying. Sometimes it performed identical to the previous model. Other times it ended up hurting the performance. In other words, adding the cues lexicon did not help performance of the model. I think that this could be due to how I implemented it though. On one of the runs I had the document `POS The program does what it should do.` change to be incorrectly classified as Neutral after adding the cues. I think this was due to the neutral class not haveing as many words added to it as the other classes. Another possible reason is that the only word in this document that was in the cues lexicon was 'should' so no word counts were really affected.

I then proceeded to train my model on the Amazon Reviews documents from 2007. This ended up taking 50 minutes on my home desktop for training and prediction. The model ended up performing well. It got a 0.850 accuracy, a 0.850 recall, a 0.850 F1-score, and a 0.851 precision.

Overall I enjoyed this assignment now that I have finished it. My original thoughts were not so positive as I was having issues regarding the classification accuracy. This was due to me accidentally trying to run the numpy argmax function on a python dictionary which has undefined behavior. Once I spent more time on the assignment, and had a better understanding of naive bayes, I found the assignment a lot more fun. I found it very interesting at how simple it was to perform sentiment analysis. The Amazon reviews dataset was particularly interesting since I felt like I created something that actually performed alright. I do not think there is anything to really change with the assignment. This is because all of the different aspects helped me better understand sentiment analysis and naive bayes. This is especially true since we were given more than one week to complete the assignment.