Spam Email Classification Using Naïve Bayes

Pavan Manjunath(109916057)
Santosh Ghosh(109770622)

Problem Definition:

We had to design a Naïve Bayes classifier in order to classify emails as spam and non-spam(ham).

Modus Operandi:

In order to classify we first calculated the prior probabilities and the unconditional probabilities of the spam and ham emails from the training data. From the training data set we observed that the ratio of spam to ham emails was 0.8. Because of this even distribution we no preprocessing of the data set was required as the data is not skewed.

For classification we used the Binary classifier coupled with Laplacian smoothing. We had also used appropriate measures to prevent underflow. The Binary classification technique for calculating the prior probabilities considers whether or not a particular word is present (1) or absent (0) in the training data set.

Additional methods used for extra credit:

For classification of spams we tried out the following methods in addition to the method mentioned above:

1. Term Frequency Classifier
2. Classifier with stop words
3. Classifier by taking into account the most commonly used spam words.
4. Classifier by taking in to account the most common spam words in found in emails

A brief Description of each process:

1. **Term Frequency Classifier:**
   This is a different approach for calculating the posterior probabilities. The ides is an email with 5 occurrences of the word like “lottery” or “lucky” is more likely to be a spam than an email with one occurrence of each. So we calculate the prior probabilities as the ratio of the number of occurrences of a word to the ratio of the total number of words in all spam emails. This approach gives us slightly better results than the binary scheme followed above.

2. **Classifier With/Without stop words:**
   Intuitively stop words like ***a,an,the,if*** etc. do not reveal much information about the nature of the email spam or not. So we tried to see how true was this intuition by taking into account and

ignoring stop words while calculating the prior probabilities. We used the standard set of stop words from nltk corpus. Interestingly the results achieved for the classification archived using stop words and not using stop words were almost same.

3. **Classification by taking into account the most common list of spam words:**
   There are certain words which are repeatedly used in emails like Lottery, Congratulations, Lucky etc Similarly many digraphs like “have won”, “pleased to” , “great opportunity”, “for free” etc. are very common. We wanted to see how well the classifier performs when we take into account these commonly used words. So we compiled a list of common spam word from the following list:

   a) http://blog.hubspot.com/blog/tabid/6307/bid/30684/The-Ultimate-List-of-Email-SPAM-Trigger-Words.aspx
   b) http://emailmarketing.comm100.com/email-marketing-ebook/spam-words.aspx
   c) http://www.forbes.com/sites/firewall/2010/03/17/the-most-common-words-in-spam-email/

   The results obtained were marginally better than the normal scheme used above.

Results:

Method for calculating prior probabilities	Precision	Recall	F-Score
Binary Classifier	90.83	93.96	92.77
Term Frequency Classification	91.28	93.65	92.44
Including Stop Words	90.25	93.15	91.67
Excluding stop words	91.15	93.02	91.54
Using common spam words	91.56	93.52	92.41

Analysis of results:

From the results obtained we see that using different features and prior probability calculation techniques does not give us very drastic changes in the performance of the classifier. It might be because of the nature of the data corpus considered which has a very even distribution of spam and non-spam emails. We think that if we used a larger dataset then the effect of the different techniques used would have amplified.