CS 4701 Practicum in Artificial Intelligence

Project Proposal

Title: Spam email filtering using neural nets

Names: Christopher Anderson (cma227) and Hao Luo (hl532)

Description:
We will train a neural network on a set of email text data to classify emails as spam or not spam. We will compare the performance of different parameters for the neural network, and time permitting will also compare the best neural network to other methods of classifying.

Approach:
The text of each email in the data set will be transformed into a list of the frequency of various common email keywords, along with whether that email is spam or not spam, so that we can treat emails numerically.
Each of the frequencies will be used as an input to the neural network, which will be trained using part of the data and the standard back-propagation algorithm. Then, the rest of the data will be used to evaluate the performance of the neural network.
We will create several neural networks and compare their performance, using different numbers of neurons, and single layer versus multilayer with different numbers of hidden layers.
If we have time after making the neural networks work, we will create a simple rule based classifier that just searches for certain hard-coded keywords, and create classifiers using other common learning algorithms - naive Bayes classifiers are commonly used for spam filtering in practice, and it would be interesting to try any other algorithms we cover, like decision trees.
The alternate algorithms will be trained and evaluated for performance in the same manner, and compared to the best performance we achieved with a neural network.

Plan:
The main way of evaluating performance will be a quantitative measurement of the percentage of emails classified correctly.

Additionally, because for spam detection it is worse to delete a good email than to let spam through, we may factor the percentage of false positives into performance.
Lastly, if possible we will evaluate the different algorithms on their time and memory usage performance.

Timeline:
November 1 - have a suitable data set of emails. Should be fairly easy to come by, though ideally we'd want data where the emails are already transformed into word frequency lists, to skip that part of the work if we can. Start implementing neural network.
November 20 - Finish implementing neural network. Run it with different parameters to get performance evaluation data.
Optionally, implement other algorithms to compare.
December 1 - be done evaluating, do the final writeup.