Ryan Vasios

This program makes use of the Natural Language Toolkit to build a Part-of-Speech (POS) tagger with Hidden Markov Models. A Hidden Markov Model, or HMM, is a statistical model assumed to be a Markov process with unobserved, or hidden, states. In our model, the states of our Markov process are parts-of-speech and the emission at any point is the word in our sentence we are attempting to tag.

To see a succinct overview of the program, examine the 'main' function beginning on line 225. It begins by separating the Penn Treebank POS Tag-Set corpus into a training set and test set. The training set consists of three thousand sentences already classified with the correct POS tags. The test set consists of sentences we will attempt to label with correct POS tags once we construct our HMM. Both training and test sets are preprocessed to include tokens for sentence beginnings, ends, and unknown tokens(ie words not appearing more than once in our training set). The first sentence of both sets, both tagged and untagged, are then printed.

We then construct and train our Hidden Markov Model which uses the training set to approximate two sets of probabilities: 'A', the probabilities of transitions between parts-of-speech, and 'B', the emission probability of a specific word in a given state. After testing the model with emitting some basic statistics, we move on to testing our model.

For comparison, we create as a baseline, a most-common part-of-speech tagger, which simply tags words with their most common POS tag. Although this basic method can obtain moderate success with tagging individual words, it doesn't contextualize taggings with transitions between parts-of-speech. By contrast, our HMM POS tagger does examine the likelihood of these transitions, using the Viterbi dynamic programming algorithm to find the most likely sequence of POS states.

We then compare the performance of both taggers on the test set. If you run the program, you will see that the HMM POS tagger outperforms the most common tagger, with 10% more correctly tagged sentences and 5% more correctly tagged words.