**LING165 Lab #2: POS tagging using first-order[[1]](#footnote-1) HMM 15-March /27-August**

**Objective of the lab**: Develop a first-order HMM see how well it works as a POS tagger (for some files from brown corpus).

Our task in this lab is to use probabilities derived from training data to then attach POS tags to test data. We’ll then compare our results to the appropriate answer file, and generate a “percentage correct” value.

**Data** consists of five sets of files—training, testing, and the answers. The training and testing files have already been partitioned using a 5-fold cross validation scheme, the result of which is a series of three sets of files:

brown.train.1 brown.test.1 brown.test.answers.1  
brown.train.2 brown.test.2 brown.test.answers.2  
brown.train.3 brown.test.3 brown.test.answers.3  
brown.train.4 brown.test.4 brown.test.answers.4  
brown.train.5 brown.test.5 brown.test.answers.5

We’ll use these 5 pre-built training files, test files, and answer files.

We’ll use Hahn’s scripts for sgt and other smoothing to come up with probabilities based on the training data, and then we’ll use Hahn’s viterbi module to create a trellis for each of our un-tagged (test) datasets.

We build our dictionaries by counting the frequencies of tag-tag sequences and tag-word associations and then from those counts, derives probabilities. We’ll capture these frequency counts in our two dictionaries, and we’ll use Hahn’s scripts to smooth the probabilities and create another dictionary of smoothed conditional probabilities.

Then we’ll have to process the “test” files (which do not have POS tags) and use the smoothed dictionary counts (pass them to Hahn’s viterbi module) to create a trellis, calculate the deltas and end up with a list named path that stores the most likely sequence of POS tags for each sentence. We do our tagging and output to a file.

**Hahn’s viterbi module:** The path data object that gets returned is a list. We step through the tags in this list and process in conjunction with our test-file data. Learn how to adapt this same technique to comparing values in our answer and output files.

Once we have the test results, we should compare to the “…answers” files and report word accuracy as a percentage of the number of words tagged correctly by the model. We’ll have 5 scores to report.

## Notes

My first attempts at this lab were a single long script that went from process to process, beginning to end, processing a single dataset. As of 26 August, I complete redesigned this script, making functions out of each separate process and invoking the functions from the menu.

### Previous Lab notes (March, April, Summer months)

Some data structure notes:

1. “First-order” means the HMM has a memory of 1 (so no history? or history of 1?) [↑](#footnote-ref-1)