Can Computers Think?
Week 9 Homework

Due Tuesday March 11

The goal of this project is to write a python program that tells you whether a given text paragraph was written by one of two authors. Let's do this step by step.

1. Download text_classification.py.
2. Download and unpack development.zip. It contains shorter text files that you can use for testing your program while you are developing it.
3. Complete function count_frequencies in text_classification.py. This function should take a list of files, read them in one by one, and calculate the frequency of all words (unigrams), the frequency of all bigrams, and the total number of words in the corpus. It should return these three things. To return more than one object, simply enumerate them with commas after the key word `return`.
   Suggestion: As a first step, only build the unigram dictionary. When that is working extend the function so that it also creates a dictionary of bigram counts. Finally, add the code necessary to compute the total number of words in the files. For the bigram dictionary, use a dictionary where the keys are tuples.
   Test your program after each step by building the unigram/bigram dictionaries for the provided test data and printing them out so that you can check them.
4. Complete the function string_prob. This function should calculate the log probability of a string based on the log probabilities of the bigrams that make up this string and the log probability of the unigram that is the first word of the string. Functions unigram_prob and bigram_prob which calculate the log probability for a unigram or bigram, respectively, are already defined. Test this function.
5. Now complete the main function. Build unigram and bigram models for two authors. Instead of just testing single passages, we want to run a whole bunch of tests. That's why I provided files with text passages. Read in these files with test passages and compute the probabilities of those passages using the models created from data by author A and the probabilities of those passages using the models created from data by author B. The function batch_test, which is already defined, will help you do this.
6. Once you are pretty sure that your program is running properly, download novels.zip and presidents.zip. Run your program with these data. Reading in and processing these files will not be instantaneous. So, don't worry if nothing seems to happen for a couple of minutes. It should not take longer than that, though.
   How well does the program do? How many passages can it classify right? How many mistakes does it make?

Background Reading

Probability for Linguists by John Goldsmith. An electronic copy is available online.

Machine Learning by Thomas G. Dietterich. An electronic copy is available online.