Methods in Computational Linguistics 2
Spring 2015

Homework #5 - Plotting and Tagging
Due at 11:59pm on May 13

For these questions, deliver the generated images, as well as the code used to create them. (First make sure that matplotlib is installed and you are able to generate any plot by recreating some of the class examples.)

1. Calculate the length of each word in brown.corpus.words(). Generate a histogram of these counts with 100 bins.

2. Generate two lists of numbers, x and y. Where x is the length of a word, and y is its frequency in nltk.corpus.brown.words(). Generate an appropriately labeled and titled scatter plot of x and y. (Use a log scale for y, the frequency of the word.)

3. Repeat question 2, but now, generate two paired lists of numbers, x/y and x2/y2, where x is the length of the word and y is its frequency in nltk.corpus.brown.words(). In this question generate x/y from function words, and x2/y2 from content words. Place both scatter plots on the same chart. Include an appropriately placed legend

4. Generate a black and white version of the previous plot.

5. Calculate the average and standard deviation of word length for NOUN, VERB, ADJECTIVE and ADVERB in the tagged brown corpus (nltk.corpus.brown), and the tagged treebank (nltk.corpus.treebank) and the tagged CoNLL corpus (nltk.corpus.conll2000). Plots these mean values using a column chart with standard deviation marks. There should be 4 groups of 3 columns each, each group of columns should be labeled by the universal part of speech label. The axes should be labeled and the chart should have a title. A legend should distinguish the 3 corpora.

Proper evaluation of a Tagger.

In class we showed a larger example using display() of how to evaluate a tagger, however, we evaluated on training data. The next set of questions asks you to perform a more proper evaluation of a tagger.

1. Write a function, based on display() that evaluates the unigram (lookup) tagger that trains on one set of sentences (from the brown corpus) and tests on a mutually exclusive set (also from the brown corpus). Plot the performance at a variety of sized models, as is currently done in display().

2. Write a modification of this function where you train on all of the brown corpus (with different sized Unigram Models), but evaluate on the conll2000 and treebank corpora. (remember to use tagset=universal)
3. Write a new function that repeats the evaluation from question 1 using a BigramTagger (with backoff to unigram and default taggers). Now the value that should be varied is the 'cutoff=n' parameter, where n is the number of observations needed for the n-gram tagger to be used. Try (at least) values from n=0…10. Plot the train set accuracy as a function of N.