Homework 6: NLTK

L555

Due Monday, October 22


Pig Latin is a simple transformation of English text. Each word of the text is converted as follows: move any consonant (or consonant cluster) that appears at the start of the word to the end, then append *ay*, e.g. *string* → *ingstray*, *idle* → *idleay*. ([http://en.wikipedia.org/wiki/Pig_Latin](http://en.wikipedia.org/wiki/Pig_Latin))

(a) Write code to convert a word to Pig Latin.

(b) Write code that converts text, instead of individual words. Use the NLTK tokenizer for this.

(c) **Bonus:** Extend it further to preserve capitalization, to keep *qu* together (i.e. so that *quiet* becomes *ietquay*), and to detect when *y* is used as a consonant (e.g. *yellow*) vs a vowel (e.g. *style*).

2. NLTK, ch. 3, #29: Readability measures are used to score the reading difficulty of a text, for the purposes of selecting texts of appropriate difficulty for language learners. Let us define $\mu_w$ to be the average number of letters per word, and $\mu_s$ to be the average number of words per sentence, in a given text. The Automated Readability Index (ARI) of the text is defined to be: $4.71 \mu_w + 0.5 \mu_s - 21.43$. Compute the ARI score for various sections of the Brown Corpus, including section *f* (popular lore) and *j* (learned). Make use of the fact that `nltk.corpus.brown.words()` produces a sequence of words, while `nltk.corpus.brown.sents()` produces a sequence of sentences.

3. Write a program that reads in the POS tagged text from file *vm.pos* (available from oncourse). Using NLTK’s FreqDist() utility, store every POS tag and its frequency.

   Note that `fdist.inc(sample)` will increment the count of *sample* in the frequency distribution `fdist`. 

   ```python
   fdist.inc(sample)
   ```