Project 2

Word Tally

Kekoa Riggin - Ling 473 - August 14, 2017

My Approach

To count and tally all the words in the corpus, I used a simple loop to read all words at a runtime of O(x). The loop opened each of the files in the corpus one at a time. To do this, I originally used a big list of all the files stored in the corpus, but I was afraid there could be errors in this list. Instead, I imported a library that allowed me to access all of the files in a directory.

Within the loop, I read the file's text into one string; I strip the string of markup, special characters, and ' at the ends of words with regular expressions; make the whole string lowercase; then I split the string into a list of words.

At this point, I used a loop, nested in the loop for files, to read each word one at a time. I stored new words into a dictionary with a count of one. I increased the count of words already stored in the dictionary by count 1.

After exiting the loop, I use a package that allows me to extract all the elements in a dictionary and put them into a sorted list by the value rather than the key. I don't know how this package works, but I know that it is far faster than any method I could have written.

Lastly, I print each element in the sorted list according to the project specifications.

Reflection

Stumbling Blocks

1. I am not an expert on runtimes and data structures. When I first had a working script, the runtime was roughly 8 minutes. After some simplification, I was at 4. It appears that my current script runs at about 2 minutes, which I am really proud of.

2. I don't feel that I was able to test this script fully. There may have been some cases that I did not think of or see in my testing process. Perhaps my lack of understanding of testing contributes to the large amount of time it takes me to do so.

3. I used more packages than I would like to. I like to use my own code because I don't like to use blackbox features. But I used two packages in this project just to make things easier. They only dealt with logistical things and contributed nothing to my tallying

(with the exception of `re`)