Writeup

Writeup: Describe what you expect to be the strengths and weaknesses of each approach. (It's okay if your guesses end up being wrong.) Try to compare them on as many axes as you can.

The algorithm in the file histogram_string_module_alg.py takes 0.040s on average to remove the punctuation for a small file (critique-of-reason.txt) and 0.780s on a large file (Ulysses-4x.txt), while the algorithm in histogram-manual-alpha-alg.py takes 0.120s for a small file (critique-of-reason.txt) and 2.300s on a large file (Ulysses-4x.txt).

This is because manually checking each word for letters of the alphabet and then removing trailing and external punctuation marks takes much more time than just calling Python's string module's punctuation on a word. Hence that is a weakness of manually removing the punctuation.

The only aspect of my code that I was able to appropriately test under the scope of this lab was the timing and because it did not take much lines of code, I included it into the appropriate files.

I would recommend the algorithm that uses Python's string module since it is quicker and more efficient.

Stretch Goal A

The algorithm in the file stretch_goal_a.py takes 0. 100s on average to remove the punctuation for a small file (critique-of-reason.txt) and 2.000s on a large file (Ulysses-4x.txt). This is because manually checking each word for punctuation marks and then removing them takes less time than checking each word for letters of the alphabet. I hence recommend this algorithm over my first manual algorithm.

References

http://matplotlib.org/1.2.1/examples/pylab_examples/histogram_demo.html

http://stackoverflow.com/questions/28473328/plot-a-histogram-of-text-values

http://stackoverflow.com/questions/25158561/ipython-pylab-print-histogram-from-dictionary

http://stackoverflow.com/questions/16050952/how-to-remove-all-the-punctuation-in-a-string-python

http://stackoverflow.com/questions/265960/best-way-to-strip-punctuation-from-a-string-in-python

https://docs.python.org/2/library/urllib.html