Matt Ruehle
SoftDes, section 2, spring 2015
Text Mining Project Writeup

Project Overview:

For this project, I used text files gleaned from the internet--for example, novel texts from Project Gutenberg, although other texts could also be used. I then sought to find the level of word equivalency/analogues between the two: much like how finding analogous gene expression in biology can be interesting when looking at organisms, I thought that looking at novels/poems/lyrics might yield interesting results here.

Implementation:

My implementation for this project was, essentially, an atypical way of analyzing the word frequencies of different texts. The ultimate, "wrapper" function takes in two text files. These files are then examined to determine their word counts and frequencies--by turning the text file into a list of words and then creating a dictionary wherein each word is mapped to its number of occurrences.

Once that's complete, the program converts these to percentages--and then goes through the various words in the text to find shared words. Whenever a shared word is found, that word's shared percentage is added to a "similarity" tracker (e.g.: if book A is 10% "apple" and book B is 8% "apple," this is seen as indicative of an additional 8% overlap). This similarity can either be looked at as a number, or -- if the program is run as the "__main__" program -- can instead be converted to a percent, rounded, and then returned in a human-readable form (as seen below).

Results:

The output of running the code as "__main__", using a couple of example comparisons:

'Alice_in_Wonderland is 61 percent equivalent to Metamorphosis (in a conceptually similar way to how humans are 80 percent genetically identical to cows).'

'The_Adventures_of_Huckleberry_Finn is 68 percent equivalent to The_Adventures_of_Tom_Sawyer (in a conceptually similar way to how humans are 80 percent genetically equivalent to cows).'

The similarity between Twain's work was actually measurably lower than I would have expected--perhaps because the two books, each being written from the narrative 1st-person perspective of a different character, necessarily had substantially different dialogue and diction.

Other texts could also be analyzed -- for example, comparing the lyrics of different artists, or the tweets of two different Twitter users (a potential novelty application) -- although I didn't find the time to actually source these and run them through yet.

Reflection:

In terms of process, I was pretty happy with the pace that I was working at -- up until ~Monday, at least. I managed incremental development pretty solidly--building sub-functions and testing them as I went to develop the project. Unfortunately, this development was cut short/rushed: I wound up contracting some manner of illness, which sapped my ability & desire to work on Software Design over the last couple of days; with the time scale of this particular project, though, I don't imagine I could've reasonably predicted or budgeted around this occurrence.

Were I to conduct this project again, I would likely test out a couple other features to see which yielded the most interesting results. Specifically, comparing the general sentiment of a novel (in terms of Pattern's sentiment analysis, at least initially) could be interesting--as could modifying the code to "disregard" words which are common amongst all literature (e.g., 'the', 'and', 'of', etc) for calculating the size of a work and the similarities.