Isabel Blancett
10.11.2017

State the Obvious

Overview

In this project, I used python to analyze similar words across different states' Wikipedia pages. More specifically, I created a program that allows users to input any two states and outputs similar words used in their descriptions. Being from a state that is hardly represented at Olin College, I came up with this idea as a way to engage my peers in learning about my state in a context that does not involve corrupt politics or bigotry. In addition, I wanted to become more comfortable with all the tools Python provides for handling strings.

Implementation

My script is composed of three major parts: content acquisition, removal of extraneous material, and comparison of states. Content acquisition was the first, and easiest, step to implement. The main step was to create a function named 'get_content' to pull their contents from Wikipedia. To test this function, I used the Olin College Wikipedia example straight from the assignment page in the format of a doctest.

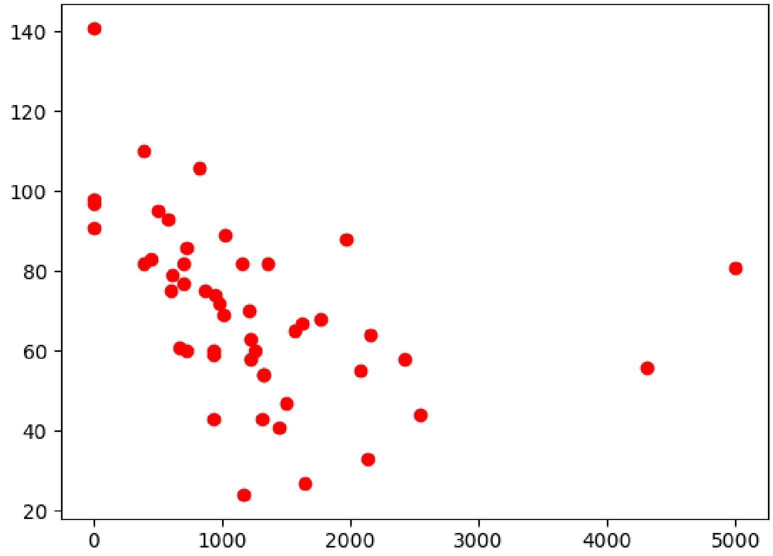
Sorting through the content proved to be much more challenging. First, I created the 'find_freq_words' function that took the name of a state and called 'get_content'. This content was then sorted by frequency and stripped of multiple occurrences and punctuation. The remaining list of words for each state were nested in a list, length of 50, named 'freq_words'. I also created a main function to loop through 'find_freq_words'. Once all states had gone through the processes mentioned above, I realized my data was going to be littered with common words like 'the' and 'is'. To fix this, I decided to remove any words that appeared in more than ten of the articles. Originally, I thought I could simply use for loops to accomplish this since I am very comfortable with them, but I soon discovered that using a dictionary would be much more efficient to compute word frequency. This was a matter of comfort versus feasibility, and as I should have assumed, feasibility won.

Finally, I had to compare the two states picked by the user. This was done in my main function by using a for and if loop to compile a master list of common words found between the two states. This listed was then outputted to the command line, or wherever the user was running the program. I also added a print statement of all the states to give me an idea of how far along the program was.

Reflection

My program is able to compare two states as picked by the user as the program. Playing around with different states triggered a question: how does distance affect the similarities of a state's culture, history, and demographics? For example, I noticed that Alabama vs. Vermont outputted a third of what Alabama vs. Mississippi outputted. Though this made sense given their proximity, I wanted to prove that my hunch that closer states had more in common. Therefore, I decided to take my project one step further.

I created a function named 'comp_al', which compares my state, Alabama, to all the other states and plots a graph of their distances vs. the number of words they had in common. I found that, indeed, on average, states closer to Alabama physically had more similar Wikipedia pages, as seen below. This isn't much of a surprise to myself, but may be a surprise to my peers, many of whom have never been closer to Alabama than 800 miles.



Distance from Alabama vs. Word Similarity

Reflection

I believe my project went very well. I adequately scoped my original project idea and carried that out well. However, I wish I had a clearer idea of an interesting final deliverable. My deliverable at it appears now was developed through my creation of the program, rather than at the beginning. I know I would have had much more direction if I had an image of this to start off with. Another thing I could have done better was my incorporation of new python techniques, such as dictionaries. Since I already am familiar with python, I have a tendency to stick to the skills that I am comfortable with, rather than branch out and add to my skill set. Had I committed to this in the beginning, I think my project could have gone a lot faster.