Text Mining (Mini Project 3)
Lauren Pudvan

Project Overview

I have recently grown an interest in educational tools for children. I was curious if I could find a correlation between the reading level of children's books and the amount of repeated words, the amount of unique words, average word lengths, and/or the tone. I used "The Very Hungry Caterpillar" by Eric Carle (easiest), "The Giving Tree" by Shel Silverstein (medium), and "The Tale Of Peter Rabbit" by Beatrix Potter (most difficult). I converted all of the books to plain text files, removed all punctuation, and changed all upper case letters to lower case in preparation for the analysis.

Implementation

The major components of my project are the different tests I mentioned above. I ran tests for the amount of times each word is used, the amount of original words, the average word length, and the tone. To perform the word frequency test I used a dictionary, because I could store the words as keys and the number of occurrences as values. I made this decision because I would not need to edit the words once they were in the dictionary, so I chose the immutable form of a dictionary. I used a list to store the word lengths in the average word length function, because I wanted something that I could easily store numbers and add to it. And, more importantly, it made it easy to take the sum of all the numbers in the list. For the tone, I used a sentiment analysis function because it was what was recommended in the project description and had good documentation to learn from.

In finding the amount of independent words I had to choose between many options. I decided to call the word frequency function and find the length of the result. I did this because I had already done the work to get all the words together with no repeats as the keys to the result of that function. At that point, I knew I would only be testing short books so it would not make much of a difference in the time it takes to run the program. If there was a chance of analyzing a long novel, I would have rewritten this code to run more efficiently and made a list of the words.

Results

Analyzing the results, I found the best way to predict the reading level of a book is by the amount of unique words. For example "The Very Hungry Caterpillar" has 112 unique words, "The Giving Tree" has 161 unique words, and "The Tale Of Peter Rabbit" has 374 unique words. All books had a neutral tone and the average word length was 4.1, 3.5, 4.2 characters (in order from lower reading level to higher), which yielded no differentiation. There was also no correlation for the word frequency. Word frequency resulted in the the most used word occurring 21, 56, and 46 times (the order is easiest to hardest book). Lastly, the most used words were character names, pronouns, and conjunctions for all the books.

One potential pit fall of this analysis is that I ran these tests for only three books. There is potential that with more testing I could find patterns in the other categories, too. Another factor is the length of each book could contribute to the differentiation of results, meaning if a book has twice the number of words, it also has twice the number of opportunities for a unique word to be used. We should try to compare only similar sized books or find some method to normalize the data.

It is not surprising that the amount of unique words correlates to the reading level of the book. As a child grows up, it is expected that their vocabulary will continue to expand. By having more unique words the child is more likely to retain their current vocabulary and add any new words to their knowledge base.

Reflection

I enjoyed the project and getting to choose a topic I was genuinely curious about. I believe my code is concise and gave me the results I desired. I could improve this project by having the results run through a correlation analysis to quantify the results and the relationships. I also could have added more books to the study in order to reinforce the results. Looking back, I feel I could have planned more for this project. I was waiting for my books to start my project, but now I know I could have started with a testing file. The testing file I ended up making was a short (only a couple lines) plain text file that allowed me to write doc-tests. I learned to find a way to start making progress earlier on in the project time line so I am less stressed later and I can get more work done.