Individual Assignment #1: Programmatically Assess the Difficulty of Various Texts

In this individual homework, you will create a program that calculates the readability of a text based on the Flesch-Kincaid reading ease formula, see here for information (https://en.wikipedia.org/wiki/Flesch%E2%80%93Kincaid_readability_tests).

Texts from historical literature can be found at Project Gutenberg((http://www.gutenberg.org/)http://www.gutenberg.org/ (http://www.gutenberg.org/)).

Flesch-Kincaid Reading Ease Formula:

$$206.835 - 1.015\left(\frac{\text{total words}}{\text{total sentences}}\right) - 84.6\left(\frac{\text{total syllables}}{\text{total words}}\right)$$

Given the above formula, you will have to figure out how to calculate the number of words, number of sentences, and number of syllables in a given document. The following table could also be codified to automatically state what level a given text is at.

Score	School level	Notes
100.00-90.00	5th grade	Very easy to read. Easily understood by an average 11-year-old student.
90.0–80.0	6th grade	Easy to read. Conversational English for consumers.
80.0–70.0	7th grade	Fairly easy to read.
70.0–60.0	8th & 9th grade	Plain English. Easily understood by 13- to 15-year-old students.
60.0–50.0	10th to 12th grade	Fairly difficult to read.
50.0–30.0	College	Difficult to read.
30.0–0.0	College graduate	Very difficult to read. Best understood by university graduates.

Creating the Program:

Your program generally should:

1. Read in a text file.
2. Parse the text file into token suitable for calculating the formula.
3. Calculate the score for the given text.

After you have completed the program, apply it to a series of texts. Some should be from famous historical literature, others from more modern sources. Perhaps look to information from online news, pop culture blogs, Twitter, etc. The sources are up to you, but I would expect at least three types of documents: books, online news stories, long-form blogs, tweets, RSS-feeds, etc.

Hint: *If your program is running slowly, consider reducing the size of the texts that you are using. Excerpts from longer texts would be fine.*

Creating the Writeup:

Answer the following questions:

1. How does the readability of literature change over the course of time? Speculate on why this occurs?
2. What differences do you notice in readability between modern sources of information? (e.g. news articles, versus long format stories, versus Twitter, etc) Are there differences in the same class of article (e.g. two news sources)? If so, why might this occur?
3. Describe a situation where you could use this algorithm to improve the

accessibility of text automatically.
4. What difficulties did you run into, if any, while working on this assignment?
5. *Optional:* Did you notice any interesting patterns or changes among various sources? Were there obvious differences? What might lead to differences? You are encouraged to elaborate on other aspects of the project or factors that were interesting to you.

Learning Objectives:

1. Demonstrate proficiency in creating a program to address the solution. (We will be programming heavily in this course.)
2. Develop familiarity with an automated scoring metric that is at the foundation of natural language processing.
3. Communicate technical concepts/issues in writing.

Rubric:

I will be looking for the following in your code: (20%)

1. Readability
2. Conciseness
3. Comments

I will grade your writeup based on the following: (80%)

1. How well were the questions addressed.
2. Clarity of writing.
3. Professionalism of the document (e.g. proper formatting, lack of typos)

Submission and Grading

Submit your *printed* writeup and a copy of your source code at the start of class in two weeks.

About Me (http://www.jaredmmoore.co

Github Repository (https://github.com/jaredmoore/Default_Course_Tempalte)