

Python Assignment: Word/Sentence Counter: Documentation

By Annemarie Schönherr

Requirements (problem description):

My python script should read an article, given as *.txt-file, and count the sentences and the words of each sentence.

First the selected *.txt-file is stored as a variable, then it will be opened and read. The result is a string-variable. By splitting this string variable at a paragraph mark like „“, „?“, „!“ one can get a list of a list containing every word of a sentence as a list item.

Based on this list, a dictionary, that takes position of each sentence (1st, 2nd, ...) as key and the number of words in this sentence contains as value, can be created. The key- and value-values of this dictionary can than be used for a barplot-diagram, that shows the number of words in every sentence.

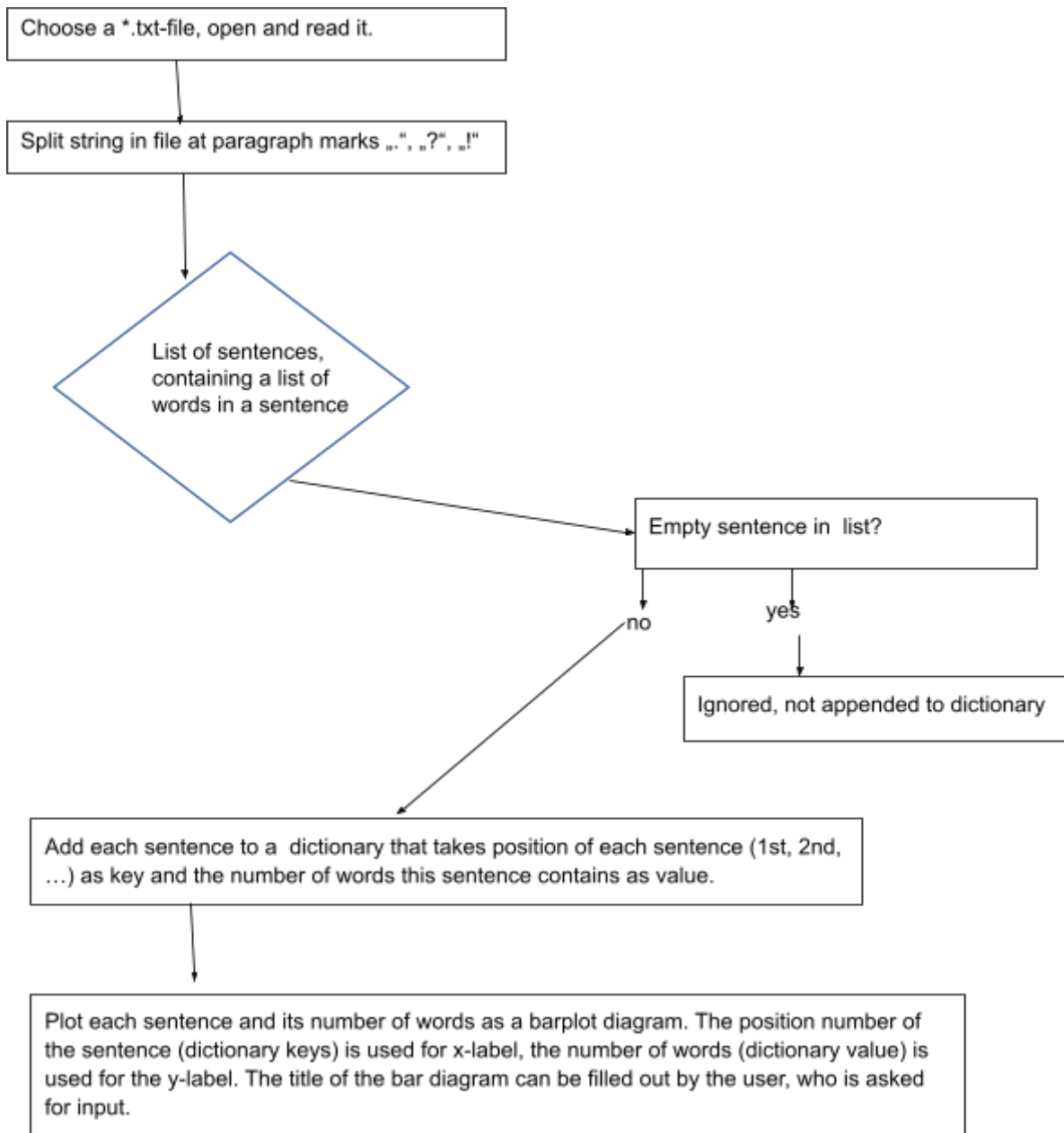
In a second part the python script should generate further statistics based on the items of the dictionary that was created in part one. These statistics generate the largest sentence (=sentence with most words), the shortest sentence (=sentence with fewest words) and the average number of words per sentence in this article. Based on the average number of words per sentence, the readability level can be stated: If the average number of words per sentence is lower than 9, the article is easy to read. If the average number of words per sentence is between 9 and 15, the article is not so easy to read. If the average number of words per sentence is between 16 and 29, the article is difficult to read. If the average number of words per sentence is higher than 30, the article is very, very difficult to read.

Further these values should be used for a summary of the statistics of the article: This summary should report the longest and the shortest sentence and the average number of words per sentence. This summary will be shown as a text table in a boxplot diagram and also be printed. The code for generating the boxplot diagram and printing summary will only be executed, if the user confirms by entering “y” or “yes” to the final question.

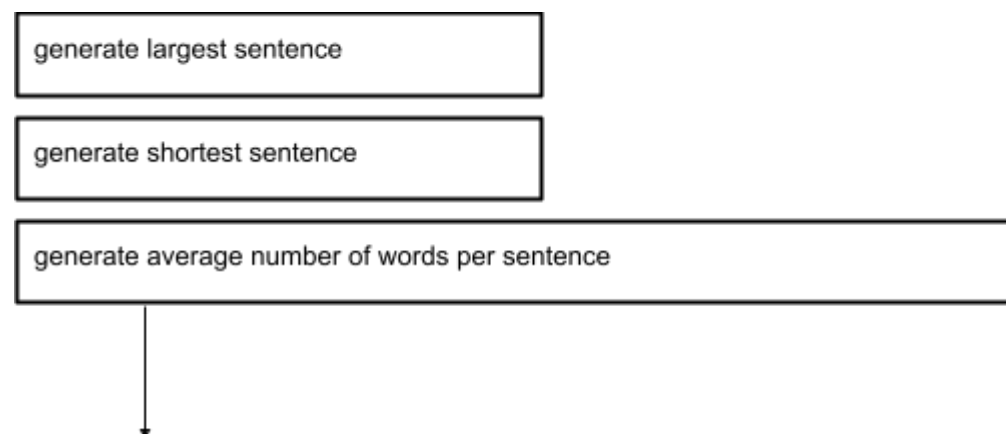
Needed libraries: matplotlib.pyplot, re,

Design:

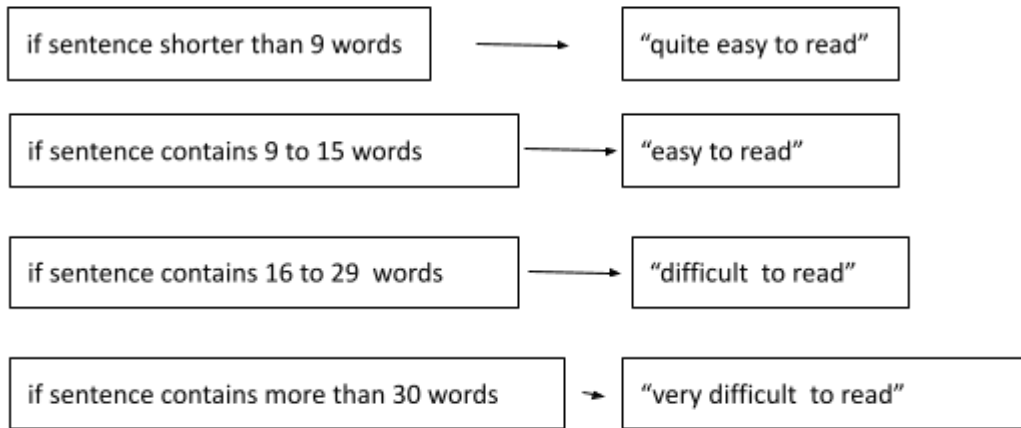
1st part:



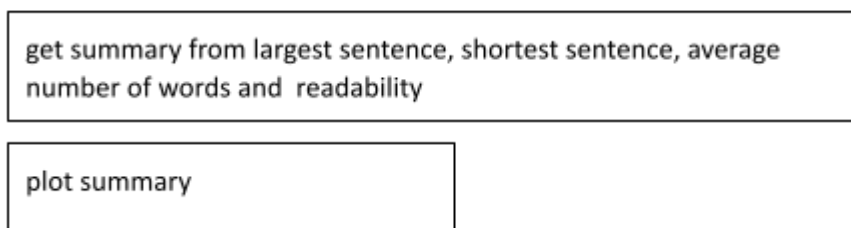
2nd part:



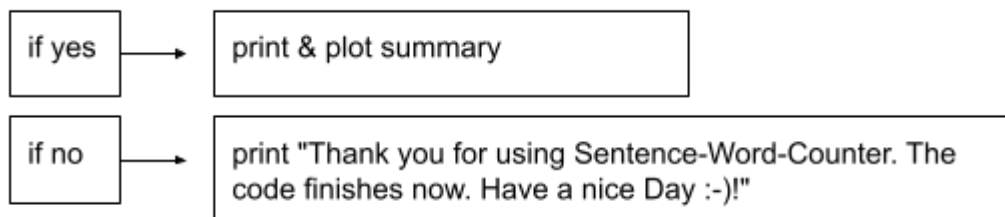
evaluate readability:



generate summary:



print and plot summary? → ask user:



Implementation:

see file "sent_word_count_plotter.py"

Verification (testing):

first testcase:

data = 'chap_1.txt'

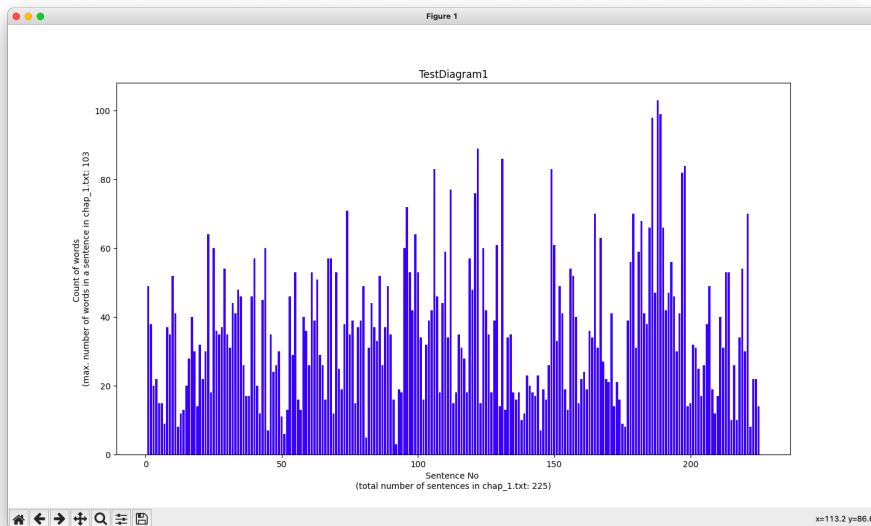
output:

“Enter a title for your diagram:”

input: “Test-Diagram1”

output:

plot:



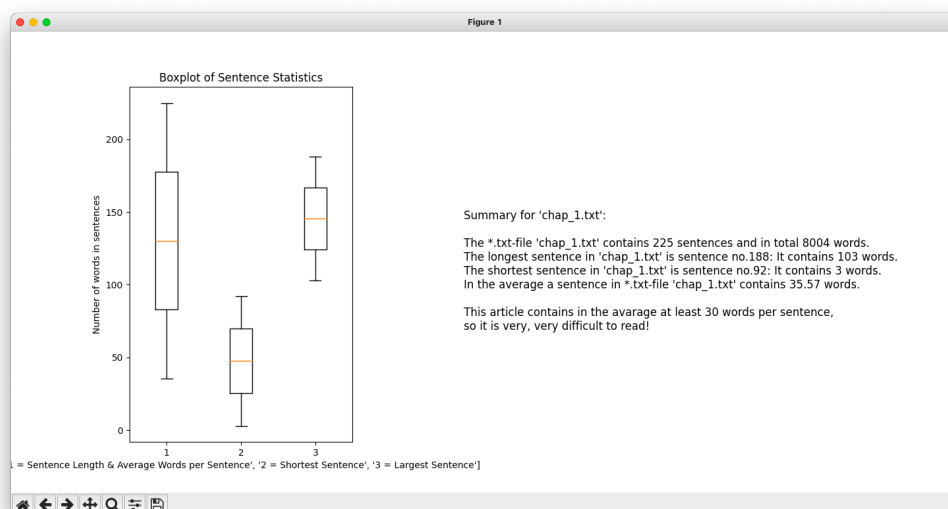
Terminal:

“Do you wish to print and plot a summary for chap_1.txt? If yes, enter 'y' or 'yes'. If not, enter any”

input: “y”

output:

plot:



Terminal:

“Summary for 'chap_1.txt':

The *.txt-file 'chap_1.txt' contains 225 sentences and in total 8004 words.

The longest sentence in 'chap_1.txt' is sentence no.188: It contains 103 words.

The shortest sentence in 'chap_1.txt' is sentence no.92: It contains 3 words.

In the average a sentence in *.txt-file 'chap_1.txt' contains 35.57 words.

This article contains in the average at least 30 words per sentence,

so it is very, very difficult to read!"

Result = Correct!

second testcase:

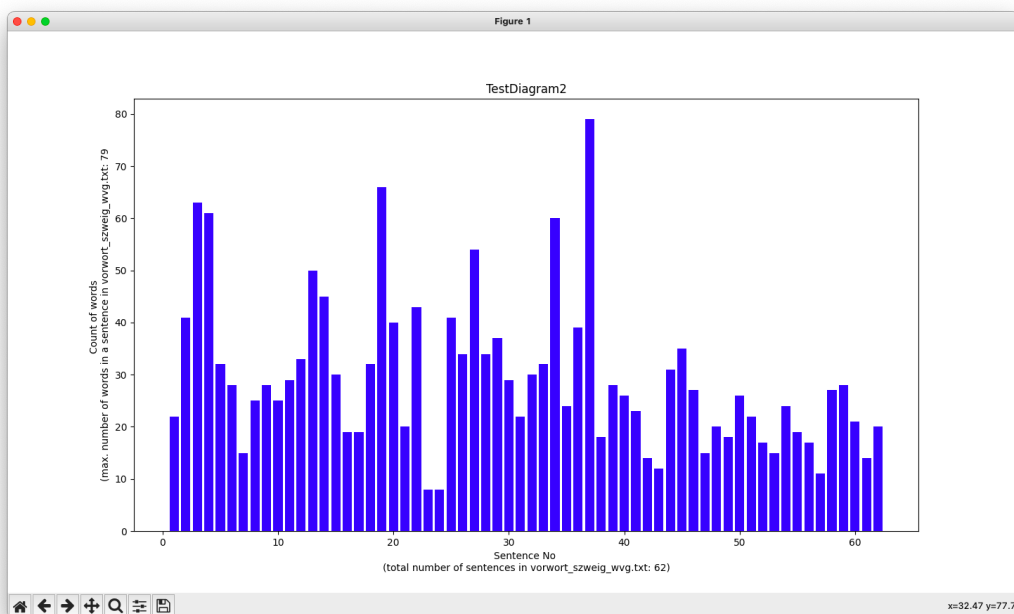
data = 'vorwort_szweig_wvg.txt'

output: "Enter a title for your diagram:"

input: "TestDiagram2"

output:

plot:



Terminal:

"Do you wish to print and plot a summary for vorwort_szweig_wvg.txt? If yes, enter 'y' or 'yes'. If not, enter any."

Input: "n"

Output: "Thank you for using Sentence-Word-Counter. The code finishes now. Have a nice Day :-)!"

Result = Correct!

Tests have been repeated with files chap_2.txt, chap_3.txt and chap_4.txt without error messages (except if there were spelling errors in the filename).