Siafakas Ilias

ITC6010A1

NATURAL LANGUAGE PROCESSING

SPRING TERM 2022

Homework1

The Homework required to analyze a random text sample and retrieve certain metrics about the number of paragraphs, sentences and words inside the text. All code was written in pure python with the exception of the sent\_tokenize() method retrieved from the NLTK toolkit

At first the program reads the .txt file that contains the text we will analyze and next the file with the stopwords that will be used later on the program. Both are assigned to two different list variables for processing. In addition the stopword file is preprocessed slightly to clear up the extra unwanted new-line characters present at the end of each word.

The readlines() method is used to split the text in paragraphs, as each paragraph ends with a new-line character. Note that the format of the text has double-spaced paragraphs, meaning that two new-line characters exist at the end of each paragraph (the same as pressing double enter in a text editor). Because of that definition the paragraphs displayed in the program appear more than in reality.

Next the program takes each paragraph and splits it into sentences using the sent\_tokenize() method. We keep the sentences in an array for further processing and add to a variable the number of sentences produced by the method. For each sentence in the nested array further processing takes place to retrieve the number of words. Each sentence is split on every space character to roughly produce the number of words. Each word array produced is added to a wordlist array to keep all the individual words.

Finally for each word produced out of the sentence, a check takes place to see if the word exists in the ‘frequency’ dictionary and increments the value by one, else it adds the new word.

This produces accurate counts of paragraphs, sentences, words and frequencies for the text file. Please note that even though “blank” paragraphs existed in the beginning they did not affect the output of the rest of the metrics.

Finally the finalized dictionary is cloned and for each word in the stopwords array is checked over the whole dictionary. If a stopword is found, then the dictionary record is deleted, which results in a cleaned dictionary with no stopwords.

The results are printed in the console at the end of the program, and the printed dictionary with stopwords follows.