USC Marshall School of Business

DSO 560: Text Analytics + NLP

Fall 2024

Homework#1

Date Given: Oct 21, 2024 Due Date: Oct 27, 2024

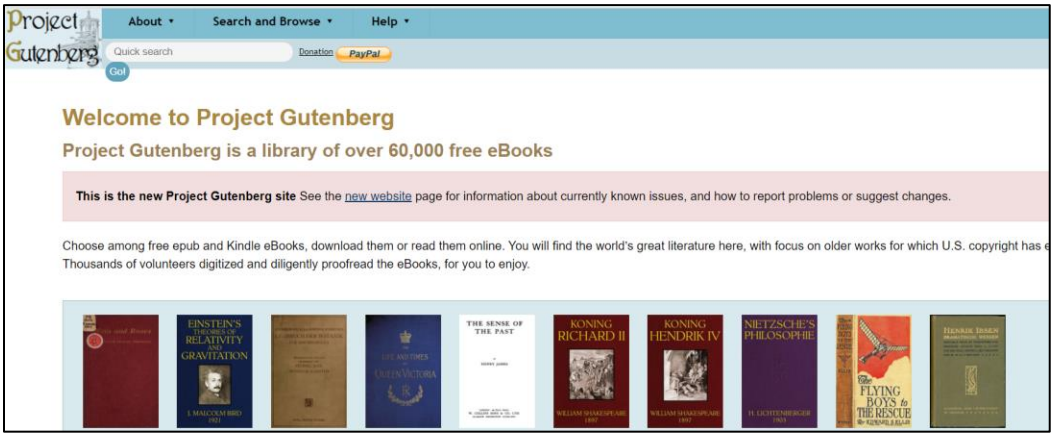
==========

There are 10 problems in this assignment.

The first 4 problems are related to Tokenization. The last 6 problems are related to analysis of text.

==========

Project Gutenberg is a library of over 60,000 free eBooks (Royalty Free).
http://www.Gutenberg.org



Download the Shakespeare's Romeo & Juliet book from the following website.
https://www.Gutenberg.org/ebooks/1513

Use TextBlob Python library for this assignment. TextBlob is an Object-oriented Natural Language Processing (NLP) library. TextBlob is built upon NLTK (Natural Language Tool Kit) and Pattern NLP libraries. TextBlob can be used to perform a variety of NLP tasks ranging from parts-of-speech tagging to sentiment analysis, and language translation to text classification.

TextBlob library is already installed in Colab. However, you need to install a few NLTK modules to make it run smoothly in Colab.

```
================================================================
from textblob import TextBlob
import nltk
nltk.download('punkt')
nltk.download('brown')
nltk.download('stopwords')
from nltk.corpus import  stopwords
================================================================
```

Create 2 TextBlobs.

1. TextBlob (blobTotal) contains the entire text of the Romeo & Juliet file.
2. TextBlob (blob1000) contains only the first 1,000 characters of the Romeo & Juliet file.

The following Python code creates 2 TextBlobs.

```
================================================================
# Read the Romeo & Juliet text.
# Retrieve only the first 1,000 characters from the text
#
textTotal = open('RomeoJuliet.txt').read()
blobTotal = TextBlob(textTotal)
#####################################
numChars = 1000
text1000 = textTotal[0:numChars+1]
blob1000 = TextBlob(text1000)
================================================================
```

Tokenization

Read the 'Romeo & Juliet.txt' file and copy the first 1,000 characters in a Python string variable 'text1000'. The data in this file is used for problem #1 - #4.

1. Find all the **word** tokens using **regular expressions** in text1000 string variable.
2. Find all the **sentence** tokens using **regular expressions** in text1000 string variable.
3. Find all the **word** tokens using **NLTK library** in text1000 string variable.
4. Find all the **sentence** tokens using **NLTK library** in text1000 string variable.

Analysis of Words

1. Count and display the **words** in the first 1,000 characters of the text. Display all the **words** by printing 10 **words** per line.
2. Count the **words** in the entire text.
3. Count the **unique words** in the entire text.
4. Count the **unique words** in the entire text after removing the **stop-words** from the list.
5. Print the top-10 **words** in the entire text with highest frequency. Also display **words'** frequency.
6. Print the top-10 **words** in the entire text with highest frequency after removing the **stop-words** from the list. Also display words' frequency.

You should expect the following answers.

Analysis of Words

1. Count and display the **words** in the first 1,000 characters of the text. Display all the **words** by printing 10 **words** per line.

```
0 . Project, 1 . Gutenberg, 2 . ’, 3 . s, 4 . Romeo, 5 . and, 6 . Juliet, 7 . by, 8 . William, 9 . Shakespeare,
10 . This, 11 . eBook, 12 . is, 13 . for, 14 . the, 15 . use, 16 . of, 17 . anyone, 18 . anywhere, 19 . in,
20 . the, 21 . United, 22 . States, 23 . and, 24 . most, 25 . other, 26 . parts, 27 . of, 28 . the, 29 . world,
30 . at, 31 . no, 32 . cost, 33 . and, 34 . with, 35 . almost, 36 . no, 37 . restrictions, 38 . whatsoever, 39 . You,
40 . may, 41 . copy, 42 . it, 43 . give, 44 . it, 45 . away, 46 . or, 47 . re-use, 48 . it, 49 . under,
50 . the, 51 . terms, 52 . of, 53 . the, 54 . Project, 55 . Gutenberg, 56 . License, 57 . included, 58 . with, 59 . this,
60 . eBook, 61 . or, 62 . online, 63 . at, 64 . www.gutenberg.org, 65 . If, 66 . you, 67 . are, 68 . not, 69 . located,
70 . in, 71 . the, 72 . United, 73 . States, 74 . you, 75 . ’, 76 . ll, 77 . have, 78 . to, 79 . check,
80 . the, 81 . laws, 82 . of, 83 . the, 84 . country, 85 . where, 86 . you, 87 . are, 88 . located, 89 . before,
90 . using, 91 . this, 92 . ebook, 93 . Title, 94 . Romeo, 95 . and, 96 . Juliet, 97 . Author, 98 . William, 99 . Shakespeare,
100 . Release, 101 . Date, 102 . November, 103 . 1998, 104 . Etext, 105 . 1513, 106 . Last, 107 . Updated, 108 . January, 109 . 30,
110 . 2019, 111 . Language, 112 . English, 113 . Character, 114 . set, 115 . encoding, 116 . UTF-8, 117 . START, 118 . OF, 119 . THIS,
120 . PROJECT, 121 . GUTENBERG, 122 . EBOOK, 123 . ROMEO, 124 . AND, 125 . JULIET, 126 . This, 127 . etext, 128 . was, 129 . produced,
130 . by, 131 . the, 132 . PG, 133 . Shakespeare, 134 . Team, 135 . a, 136 . team, 137 . of, 138 . about, 139 . twenty,
140 . Project, 141 . Gutenberg, 142 . volunteers, 143 . THE, 144 . TRAGEDY, 145 . OF, 146 . ROMEO, 147 . AND, 148 . JULIET, 149 . by,
150 . William, 151 . Shakespeare, 152 . Contents, 153 . THE, 154 . PROLOGUE, 155 . ACT, 156 . I, 157 . Scene, 158 . I, 159 . A,
160 . public, 161 . place, 162 . Scene, 163 . II, 164 . A, 165 . Street, 166 . Sc,

Count of words =  167
```

2. Count the words in the entire text.

```
Count of words =  30796
```

3. Count the **unique words** in the entire text.

```
Total number of unique words in the text = 4145
Total number of words in the text = 30796
```

4. Count the unique words in the entire text after removing the stop-words from the list.

```
Total number of unique words in the text = 4145
Total number of unique words in the text AFTER removing stop wprds = 4017
Total number of stop words in the text = 128
```

5. Print the top-10 words in the entire text with highest frequency. Also display words' frequency.

	word	frequency
0	the	876
1	’	869
2	and	808
3	i	655
4	to	626
5	a	542
6	of	519
7	in	395
8	is	372
9	that	369

6. Print the top-10 words in the entire text with highest frequency after removing the stop-words from the list. Also display words' frequency.

	word	frequency
0	’	869
1	romeo	320
2	thou	278
3	juliet	195
4	thy	170
5	capulet	163
6	nurse	149
7	love	148
8	thee	138
9	lady	117