Lab#1, NLP Spring 2023

This is due on 2023/03/06 15:30, commit to your github as a PDF (lab1.pdf) (File>Print>Save as PDF).

IMPORTANT: After copying this notebook to your Google Drive, please paste a link to it below. To get a publicly-accessible link, hit the *Share* button at the top right, then click "Get shareable link" and copy over the result. If you fail to do this, you will receive no credit for this lab!

LINK: paste your link here

https://drive.google.com/file/d/1d7Ryo0munni10RkjuC2j99K_nl7dkkZ2/view?usp=share_link

---

Student ID: B0928010

Name: 陳柏曄

Question 1 (100 points)

Let's switch over to coding! Write some code in this cell to compute the number of unique word **tokens** in this paragraph (5 steps of Text Normalisation: 1. Lowercase Conversion, 2. Remove punctuations, 3. Stemming, 4. Lemmatisation, 5. Stopword Removal). Use a whitespace tokenizer to separate words (i.e., split the string by white space). Be sure that the cell's output is visible in the PDF file you turn in on Github.

```
import nltk
```

```
paragraph = '''Last night I dreamed I went to Manderley again. It seemed to me
that I was passing through the iron gates that led to the driveway.
The drive was just a narrow track now, its stony surface covered
with grass and weeds. Sometimes, when I thought I had lost it, it
would appear again, beneath a fallen tree or beyond a muddy pool
formed by the winter rains. The trees had thrown out new
low branches which stretched across my way. I came to the house
suddenly, and stood there with my heart beating fast and tears
filling my eyes.'''

# DO NOT MODIFY THE VARIABLES
tokens = 0
word_tokens = []


# YOUR CODE HERE! POPULATE THE tokens and word_tokens VARIABLES WITH THE CORRECT

nltk.download("punkt")

token_list = nltk.word_tokenize(paragraph)
token_list = [token.lower() for token in token_list]
```

```
    [nltk_data] Downloading package punkt to /Users/jeffrey/nltk_data...
    [nltk_data]   Package punkt is already up-to-date!
```

```
def remove_punkt(token):
    return [word for word in token if word.isalpha()]

token_list = remove_punkt(token_list)


nltk.download("stopwords")

stop_words = set(nltk.corpus.stopwords.words("english"))

token_list = [token for token in token_list if token not in stop_words]
```

```
    [nltk_data] Downloading package stopwords to
    [nltk_data]     /Users/jeffrey/nltk_data...
    [nltk_data]   Package stopwords is already up-to-date!
```

```
port = nltk.stem.porter.PorterStemmer()
lanc = nltk.stem.lancaster.LancasterStemmer()
snow = nltk.stem.SnowballStemmer("english")

token_list = [port.stem(token) for token in token_list]
token_list = [lanc.stem(token) for token in token_list]
token_list = [snow.stem(token) for token in token_list]
```

```
nltk.download('wordnet')

lemmatizer = nltk.stem.WordNetLemmatizer()
token_list = [lemmatizer.lemmatize(token) for token in token_list]
```

```
[nltk_data] Downloading package wordnet to /Users/jeffrey/nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
```

```
tokens = len(token)
word_tokens = list(token)

# DO NOT MODIFY THE BELOW LINE!
print('Number of word tokens: %d' % (tokens))
print("printing lists separated by commas")
print(*word_tokens, sep = ", ")
```

```
Number of word tokens: 111
printing lists separated by commas
Last, night, I, dreamed, I, went, to, Manderley, again, ., It, seemed, to,
```