Lab#1, NLP Spring 2025

This is due on 2025/03/10 16:00, 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://colab.research.google.com/drive/xxxxxxxxx

Student ID:

Name:

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.

按兩下 (或按 Enter 鍵) 即可編輯

```
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 = []

import nltk
from nltk.stem import PorterStemmer, LancasterStemmer, SnowballStemmer, WordNetLemmatizer
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize


nltk.download("stopwords")
nltk.download("punkt")
nltk.download("wordnet")

# 1. 小寫轉換（Lowercase Conversion）
word_tokens = word_tokenize(paragraph)
tokens_lower = [token.lower() for token in word_tokens]

# 2. 移除標點符號（Remove punctuations）
def remove_punct(tokens):
        return [word for word in tokens if word.isalpha()]

clean_tokens = remove_punct(tokens_lower)

# 3. 語幹提取（Stemming）
porter = PorterStemmer()
stemmed_port = [porter.stem(token) for token in clean_tokens]

lanc = LancasterStemmer()
stemmed_lanc = [lanc.stem(token) for token in clean_tokens]

snow = SnowballStemmer("english")
stemmed_snow = [snow.stem(token) for token in clean_tokens]


# 4. 詞形還原（Lemmatisation）
lemmatiser = WordNetLemmatizer()
lemmatised = [lemmatiser.lemmatize(token) for token in stemmed_snow]

```

```
# 5. 停用詞去除（Stopword Removal）
stop_words = set(stopwords.words("english"))
words_no_stop = [word for word in lemmatised if word not in stop_words]

word_tokens = set(words_no_stop)

tokens = len(word_tokens)


# 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: 51
printing lists separated by commas
sometim, new, way, thrown, night, pool, beyond, driveway, rain, surfac, fallen, grass, stoni, cover, form, across, sudden, heart, came, winter, l
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package wordnet to /root/nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
```