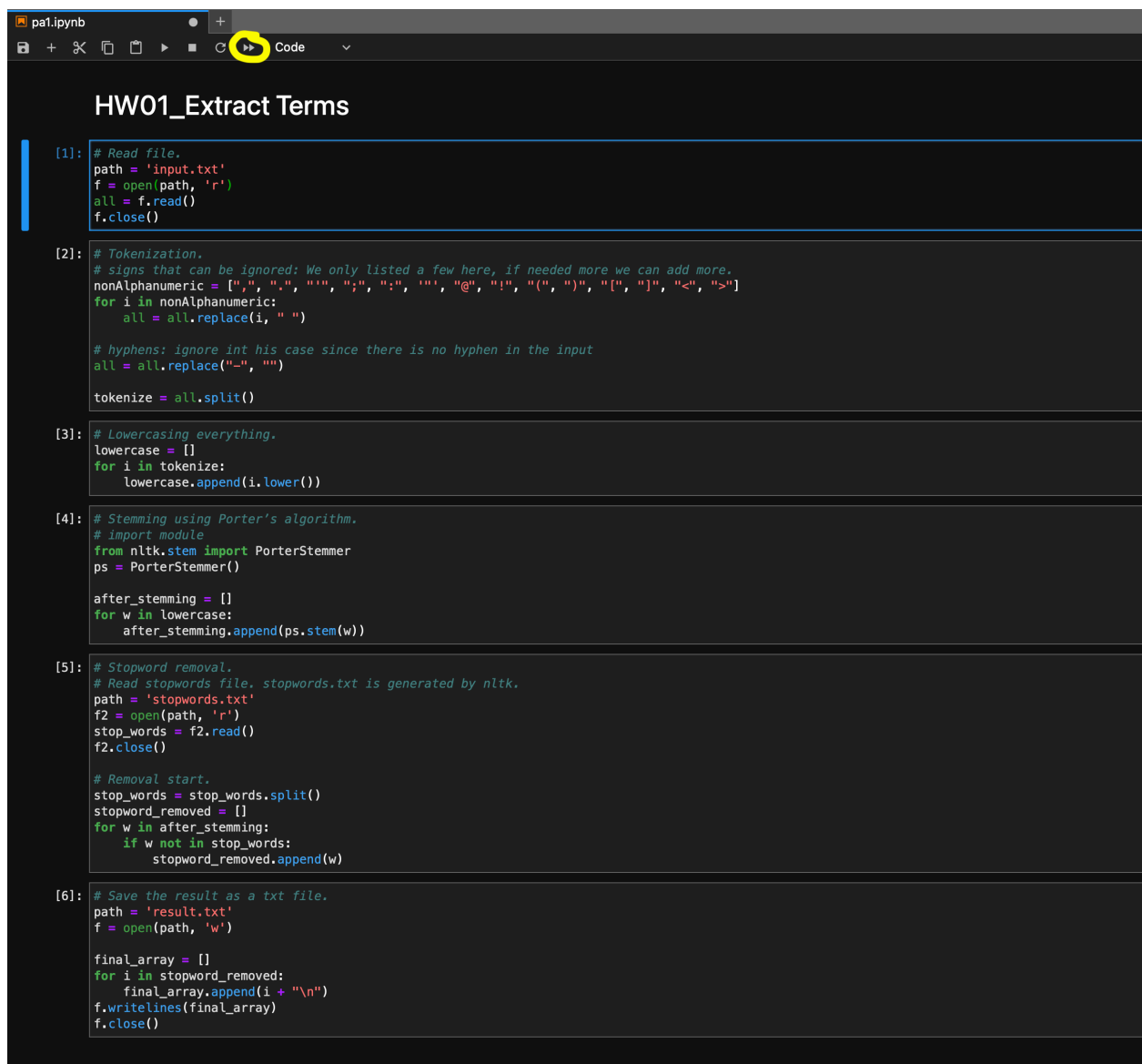


1. 執行環境：Jupyter Lab
2. 程式語言：Python 3.10.5
3. 執行方式：

方法一、使用 pa1.ipynb 檔

- (1) 使用 Jupyter Lab 開啟 pa1.ipynb 檔
- (2) pip3 install nltk 用於 Porter's Algorithm
- (3) 將整個檔案全部執行即可



```
[1]: # Read file.
path = 'input.txt'
f = open(path, 'r')
all = f.read()
f.close()

[2]: # Tokenization.
# signs that can be ignored: We only listed a few here, if needed more we can add more.
nonAlphanumeric = [" ", ".", ",", "!", ":", ";", "'", "@", "(", ")", "[", "]", "<", ">"]
for i in nonAlphanumeric:
    all = all.replace(i, " ")

# hyphens: ignore int his case since there is no hyphen in the input
all = all.replace("-", " ")

tokenize = all.split()

[3]: # Lowercasing everything.
lowercase = []
for i in tokenize:
    lowercase.append(i.lower())

[4]: # Stemming using Porter's algorithm.
# import module
from nltk.stem import PorterStemmer
ps = PorterStemmer()

after_stemming = []
for w in lowercase:
    after_stemming.append(ps.stem(w))

[5]: # Stopword removal.
# Read stopwords file. stopwords.txt is generated by nltk.
path = 'stopwords.txt'
f2 = open(path, 'r')
stop_words = f2.read()
f2.close()

# Removal start.
stop_words = stop_words.split()
stopword_removed = []
for w in after_stemming:
    if w not in stop_words:
        stopword_removed.append(w)

[6]: # Save the result as a txt file.
path = 'result.txt'
f = open(path, 'w')

final_array = []
for i in stopword_removed:
    final_array.append(i + "\n")
f.writelines(final_array)
f.close()
```

方法二、使用 pa1.py 檔

- (1) pip3 install nltk 用於 Porter's Algorithm
- (2) 直接用 python3 執行

4. 作業處理邏輯說明：

(1) Read file: Download file from the source provided in the homework document and save it as "input.txt". Use `open()` read only to read file and `close()` to finish.

(2) Tokenization: Craft a list with signs we want to split as `nonAlphanumeric`. Then, replace the signs with blank and eliminate the hyphens. Finally, use `".split()"` to split string in to a list of words whenever the next element is a `"\n"` or a blank.

(3) Lowercasing everything: Use `".lower()"` to lower case everything.

(4) Stemming using Porter's algorithm: Use the api `"from nltk.stem import PorterStemmer"` allowed by the professor for stemming words.

(5) Stop word removal: Read file `"stopwords.txt"`, which is generated by `"nltk"` (approved by the professor). Whenever the word after stemming is not a stop word, we keep it to the next section.

(6) Save the results as an txt file: Finally, we save the result in the previous section (5) into file `"result.txt"`.