Document for Programming Assignment 1

R06725016 吳亞璇
2017/10/03

程式語言

- Python 3.5.2

執行環境

- Ubuntu 16.04.3 LTS
  - LTS 為 Long Term Support

執行方式

- 安裝 Python

```
sudo apt-get install python3.5
```

- 檢查 Python 版本

```
$ python3 --version
Python 3.5.2  # 或任何 3.5.2 以上的版本
```

- 先建立一個名為 `hw1` 的資料夾（假設要執行的程式會被放在 hw1 資料夾下）

```
mkdir hw1
```

- 切換至剛剛建立的目錄

```
cd hw1
```

- 安裝虛擬環境（virtual environment）[註一]

```
sudo apt-get install python3-venv
```

- 建立虛擬環境（virtual environment）

```
python3 -m venv hw1_venv
```

- 執行虛擬環境

```
$ source hw1_venv/bin/activate
(hw1_venv) ~/hw1$  # 無論所在路徑為何，(hw1_venv)必會出現，表示你已經在虛擬環境中
```

- 安裝 Python 套件 — nltk [註二]

```
pip3 install nltk
```

- 下載 nltk 的 stopwords

```
$ python3
>>> import nltk
>>> nltk.download('stopwords')  # 等待下載完成
```

```
>>> exit()
```

- 執行程式

```
python3 hw1.py  # 輸出為 result.txt
```

[註一] 使用虛擬環境有許多優點，包括：

- 讓專案有獨立的執行環境。當多人在不同機器上跑同一專案時，也能確保環境的一致性。
- 便於控管套件，避免升級套件時影響到其他專案的執行。

[註二] NLTK（Natural Language Toolkit）

- NLTK 是 Python 的自然語言處理套件，附帶不同程度的預先處理功能，例如：Tokenization
- 此次作業中我只用 NTLK 來實作 Porter's Stemming Algorithm

文件處理邏輯說明

處理流程分為三個階段，依序為 Tokenization, Normalization 和 Stemming

- Tokenization
  - 移除 tab 和換行
  - 移除 `'s`
  - 移除標點符號
  - 以 `space` 將文件內容切割，得到 **tokens**
- Normalization
  - 將每個 token 轉為小寫
  - 移除重複的 token
  - 移除屬於 `stop words` 的 token，得到 **terms**
- Stemming
  - 使用 nltk 的 PorterStemmer 對 terms 做 stemming
  - 得到處理結果，寫檔