Python 網路期末專題

陳英翔

主題 ：PTT 政黑板的文章爬取與資料分析

目標

- 針對 PTT 政黑板爬取出的文章，透過 jieba 將其文字內容做拆解
- 過濾 stopwords，針對經常出現的關鍵字做排名
- 將結果以文字雲方式呈現
- 從爬取的內容分析來自相同 IP 不同帳號的文章，列出有網軍嫌疑的帳號
- 針對有疑慮的帳號做詞類分布的分析
- 分析指定帳號在特定期間的活動情形

程式碼介紹 & 成果展示

程式碼放置處
個人 Github: https://github.com/0307eito/2nd-PyCrawlerMarathon

(一)爬取資料

使用 requests 和 BeautifulSoup 對 PTT 政黑版進行爬蟲，爬取前 50 頁所有文章(約 1000 篇)。 我使用的方法是將所有 1000 篇文章的網址歸納至 list 內，再用迴圈的方式一個個提取並去載入其內文，去做爬取的動作。

1. 爬取出 PTT 政黑版前 50 頁的頁面網址。

```
import requests
from bs4 import BeautifulSoup

url = 'https://www.ptt.cc/bbs/HatePolitics/index.html'
my_headers = {'cookie': 'over18=1;'}
r = requests.get(url, headers = my_headers)
soup = BeautifulSoup(r.text, "html5lib")

url_list=['https://www.ptt.cc/bbs/HatePolitics/index.html']

a=soup.find_all('a',class_="btn wide")
second_page=a[1]['href']
second_page_number=re.findall('[0-9]+', second_page)
count=int(second_page_number[0])+1

for i in range(49):
    count-=1
    url_list.append('https://www.ptt.cc/bbs/HatePolitics/index'+str(count)+'.html')
```

2. 接下來爬出 50 個頁面內的所有文章網址(一個頁面約有 20 篇文章)。if 文是為了跳過已被刪除的文章。

```
title_url=[]

for i in range(0,50):
    r=requests.get(url_list[i],headers=my_headers)
    soup = BeautifulSoup(r.text, "html5lib")
    b=soup.find_all('div',class_="title")

    for i in range(len(b)):
        if b[i].text.replace('\n','').replace('\t','')[0:8]!='(本文已被刪除)' and
b[i].text.replace('\n','').replace('\t','')[0:3]!='(已被':
            title_url.append('https://www.ptt.cc'+b[i].a['href'])
```

3. 上一步取得了所有文章的網址，現在開始爬取其內文。因為字數量龐大，我以文字檔做儲存。

```
for i in range(len(title_url)):
    url = title_url[i]
    r = requests.get(url, headers = my_headers)
    soup = BeautifulSoup(r.text, "html5lib")

    a=soup.find_all('div', class_="bbs-screen bbs-content")
    b=soup.find_all('span',class_='article-meta-value')
    if len(b)==4:
        file = open('test.txt', 'a',encoding="utf-8")
        file.write(str(a[0].contents[4]).replace('\n','').replace(' ',''))
        file.close()
```

(二)文字雲製作 & 關鍵字分析與統整

接下來利用 jieba 進行中文斷詞，過濾 stopwords，再利用 counter 計數函數統計各個關鍵字的出現次數，用長條圖以及文字雲的方式去呈現結果。

1. 首先設定好要讀取的文字檔以及停用詞資料(停用詞是參考下方連結)，用 jieba 拆解文章為字詞，再利用 Counter 統計數量。

https://github.com/tomlinNTUB/Python-in-5-days

```
import jieba
import jieba.analyse
from wordcloud import WordCloud
import numpy as np
from collections import Counter

text=open('test.txt','r',encoding="utf-8").read()
jieba.set_dictionary('字庫.txt')

with open('停用詞0.txt','r',encoding="utf-8-sig") as f:
    stops=f.read().split('\n')

terms=[]
for t in jieba.cut(text,cut_all=False):
    if t not in stops:
        terms.append(t)

diction=Counter(terms)
```

2. 將 Counter 統計結果直接代入 WordCloud 函數，做出文字雲。我個人另外使用了圖像庫 PIL 用來做出喜好的形狀。注意要記得指定中文字型，不然文字雲會以亂碼呈現。

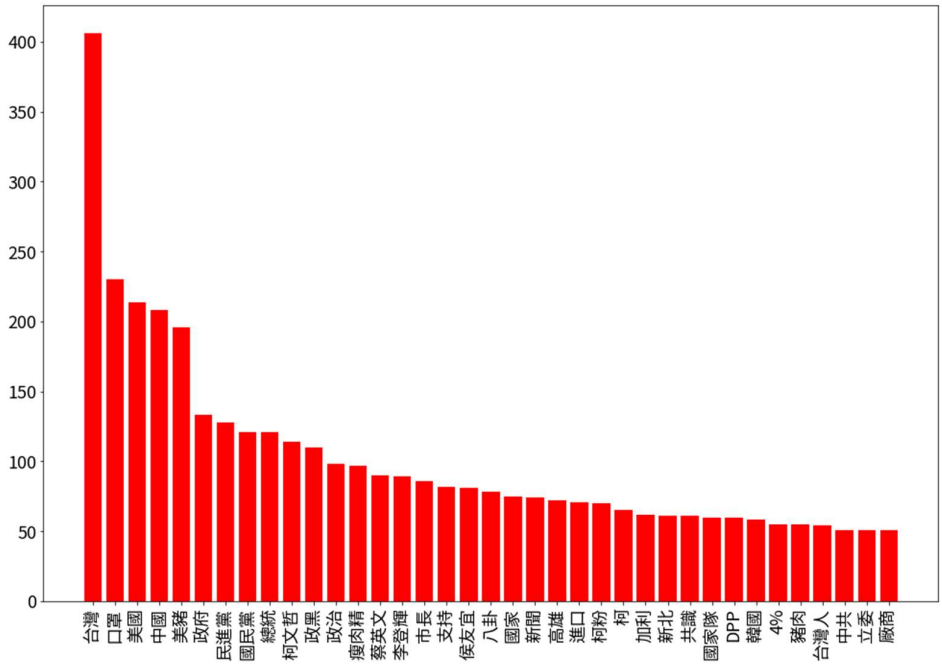
```
from PIL import Image

font='C:\WINDOWS\FONTS\MSJHBD.TTC'

mask=np.array(Image.open('hand.jpg'))
wordcloud=WordCloud(width=900,height=800,background_color='white',font_path=font,m
ask=mask,max_words=200)
wordcloud.generate_from_frequencies(diction)

plt.figure(figsize=(25,25))
plt.imshow(wordcloud)
plt.axis('off')
plt.show()
```

3. 統計出的結果是以 Counter 結構構成，這邊為了後續處理方便，先將 Counter 結構轉為字典形式。然後從轉換過後的字典中取出出現次數超過 50 的字詞，重新定義一個新的字典。

```
keys=[]
values=[]
for key in diction.keys():
    keys.append(key)
for value in diction.values():
    values.append(value)
dic_all=[]
dic=dict(zip(keys,values))
for i in range(len(keys)):
    for ii in range(dic[str(keys[i])]):
        dic_all.append(keys[i])
k=[]
v=[]
for e in dic.values():
    if e>50:
        v.append(e)
for d in dic:
    if dic[str(d)]>50:
        k.append(d)
dic_50=dict(zip(k,v))
```

4. 將字典的 keys 和 values 輸出為 list 形式，用 sort 函數由出現次數多到少做排序，變可直接做出長條圖來呈現關鍵字的排名。

```
kk=[]
vv=[]
for k, v in sorted(dic_50.items(), key=lambda x: -x[1]):
    kk.append(str(k))
    vv.append(v)
import matplotlib.pyplot as plt
plt.rcParams['font.sans-serif'] = ['Taipei Sans TC Beta']
plt.rcParams["font.size"] = 18
plt.figure(figsize=(14, 10))
plt.bar(list(kk), vv, color="r",width=0.8)
plt.xticks(rotation=90)
plt.tight_layout()
plt.show()
```



(三)帳號、IP 的分析

我針對爬出的前 500 篇文章作分析。上面已經爬出所有文章的網址了，現在使用相同的方式去爬取標題、帳號、IP 等資訊，用 pd 的 DataFrame 資料結構做整理。 接下來藉由整理的結果，過濾來自相同 IP 不同帳號的文章，針對此類帳號的詞類分布、活動情形做分析。

1. 使用 requests 和 BeautifulSoup 爬出所有文章的標題與發文者帳號名稱。

```
title=[]
for i in range(0,50):
    my_headers = {'cookie': 'over18=1;'}
    r=requests.get(url_list[i],headers=my_headers)
    soup = BeautifulSoup(r.text, "html5lib")
    b=soup.find_all('div',class_="title")
    for i in range(len(b)):
        if b[i].text.replace('\n','').replace('\t','')[0:8]!='(本文已被刪除)' and
b[i].text.replace('\n','').replace('\t','')[0:3]!='(已被':
            title.append(b[i].text.replace('\t','').replace('\n',''))
user=[]
for i in range(0,50):
    my_headers = {'cookie': 'over18=1;'}
    r=requests.get(url_list[i],headers=my_headers)
    soup = BeautifulSoup(r.text, "html5lib")
    b=soup.find_all('div',class_="title")
    for i in range(len(b)):
        if b[i].text.replace('\n','').replace('\t','')[0:8]!='(本文已被刪除)' and
b[i].text.replace('\n','').replace('\t','')[0:3]!='(已被':
            c=soup.find_all('div',class_="author")
            user.append(c[i].text)
```

2. 此處爬取 IP 的方法與上述爬取文章的方式一樣。倒數第二行的 re 正規表達函數是為了要過濾爬取出的文字做使用。

```
user_ip=[]
my_headers = {'cookie': 'over18=1;'}

for i in range(500):
    url = title_url[i]
    r = requests.get(url, headers = my_headers)
    soup = BeautifulSoup(r.text, "html5lib")
    b=soup.find_all('span',class_="f2")
    for i in range(len(b)):
        a=b[i].text
        if a[0:27]=='※ 發信站: 批踢踢實業坊(ptt.cc), 來自: ':
            c=re.search(r'([ ](\d{1,3}).(\d{1,3}).(\d{1,3}).(\d{1,3})[ ])',a)
            user_ip.append(c.group().replace(' ',''))
```

3. 將上面爬出的標題、帳號、IP 等資料限定為前 500 筆，使用 pd 的 DataFrame 資料結構做整理。

```
s=pd.DataFrame({'標題': title[0:500],
                '作者': user[0:500],
                'IP': user_ip[0:500]})

display(s)
```

	標題	作者	IP
0	[新聞] 抓到加利靠「藥師」 蔡英文、蘇貞昌、陳	Tiffwyetha	1.161.120.162
1	[討論] 其實整人政治立場大多偏向哪裡？	jordanlove	27.242.7.166
2	[討論] CF有哪個週末在台北市跑行程嗎？	nogood	123.241.151.6
3	[新聞] 義民祭與賴清德同台 鄭文燦：我們感情好有	ToHsiang	101.136.30.55
4	[黑特] 新聞說國軍都不吃美豬	nbarepeat	114.34.156.22
...	...	...	...
495	[討論] 主流民意版都不管新竹縣喝屍水欸	goldenfire	39.9.163.244
496	[討論] 為何賴清德這麼不得民進黨青睞？	goodgooddad	27.242.140.79
497	Re: [討論] 討論一下版規	jacklyl	180.217.181.2
498	[黑特] 2022，國民黨將收復失土臺南市	brella	114.198.160.22
499	[黑特] 如果版規禁4%仔，該怎麼稱呼柯糞？	growl	180.217.172.123

4. 過濾出出現次數大於 1 的 IP。

```
c = collections.Counter(user_ip)
ip=[]
ip_=[]
b=c.most_common(200)
for i in range(len(b)):
    if b[i][1]>1:
        ip.append(b[i][0])
        ip_.append(b[i][1])

p=pd.DataFrame({'IP': ip,
                '數量': ip_})
display(p)
```

	IP	數量
0	150.116.179.47	6
1	49.216.161.56	6
2	39.12.137.48	6
3	122.118.108.10	6
4	61.222.53.13	5
...	...	...
96	39.9.192.210	2
97	125.227.186.108	2
98	27.242.222.11	2
99	42.73.36.177	2
100	101.14.194.130	2

5. 上面的表格可以看出出現次數最多的 IP 有四組，數量都為 6 次，此處針對這四組 IP 做分析。

```
main_ip=[]
b=c.most_common(200)
for i in range(len(b)):
    if b[i][1]>5:
        main_ip.append(b[i][0])
def ip(iii):
    bb=[]
    cc=[]
    dd=[]
    a=[i for i, x in enumerate(user_ip[0:500]) if x == main_ip[iii]]
    for i in range(len(a)):
        bb.append(user[0:500][a[i]])
    for i in range(len(a)):
        cc.append(user_ip[0:500][a[i]])
    for i in range(len(a)):
        dd.append(title[0:500][a[i]])
    p=pd.DataFrame({'title':dd,
                    'Author':bb,
                    'IP': cc})
    display(p)

a=len(main_ip)
for i in range(a):
    ip(int(i))
```

	title	Author	IP
0	Re: [新聞] 到後山拔樁？民眾黨花蓮總服務處成立　主	linchadwick	49.216.161.56
1	[新聞] 學生、軍警不吃美豬！陳時中苦笑：我沒說	nightwing	49.216.161.56
2	[新聞] 國民黨：現金用光 盼年底前籌措1億2000萬	heinse	49.216.161.56
3	Re: [討論] 為什麼柯黃都從c-chat出來居多	pkpkc	49.216.161.56
4	[討論] 台北市立交響樂團吹中國喇叭	Ophiuchus	49.216.161.56
5	Re: [新聞] 柯文哲自豪零舉債還款570億　點名外縣市	sunyeah	49.216.161.56

	title	Author	IP
0	[公告] 政治黑特板板規_20200523_V6.0	Rrrxddd	150.116.179.47
1	[公告] 關於轉文至八卦版	kero2377	150.116.179.47
2	[公告] #1VK8zpCe tw689 版規2-2 21天	phoenixzero	150.116.179.47
3	[討論] 中、印是不是快開打了？	ppp123	150.116.179.47
4	[討論] 廢物堆積場	takashi01	150.116.179.47
5	[新聞] 加利老闆怒嗆國家隊 羅友志轟「酸更」：	tenfu	150.116.179.47

	title	Author	IP
0	[新聞] 到後山拔樁？民眾黨花蓮總服務處成立　主	ice80712	39.12.137.48
1	Re: [轉錄] #需兒碎碎唸	wupaul	39.12.137.48
2	[黑特] 陳時中：予豈好嘮頭食美豬哉？予不得已也	NuCat	39.12.137.48
3	[討論] 政黑彈總寫綠彈總都是雞碎吧	berkeley5566	39.12.137.48
4	Re: [討論] 為什麼柯黃都從c-chat出來居多	leptoneta	39.12.137.48
5	[新聞] 劣質中國製口罩退入費名制 侯友宜：全面	knnji	39.12.137.48

	title	Author	IP
0	[新聞] 加利負責人350萬元交保 侯友宜：一定要從	rockawii	122.118.108.10
1	Re: [Live] 美豬ractopamine容許量記者會	devidevi	122.118.108.10
2	Re: [討論] 文宣部被人家公幹 整天做馊魚米粉!	ericisfish	122.118.108.10
3	[Live] 搜克天團要回家了	stantheman	122.118.108.10
4	[新聞] 台美日歌整團供應雞聯合聲明 確保供應雞	nicetree	122.118.108.10
5	Re: [討論] 有發現政黑被入侵嗎	zeuswell	122.118.108.10

6. 我從上面隨機選出了兩組帳號(ice80712、rockawii)，針對這兩組帳號的詞類分布以及活動情形做分析和比較。 使用的方式是，分別爬出這兩組帳號的所有文章，以文字檔做儲存後用 jieba 進行中文斷詞，過濾 stopwords，再將出現次數高的字詞用長條圖去呈現。

爬出所有文章網址

```
user_name=['ice80712','rockawii']

url = 'https://www.ptt.cc/bbs/HatePolitics/search?page=1&q=author%3A'+user_name[0]
my_headers = {'cookie': 'over18=1;'}
r = requests.get(url, headers = my_headers)
soup = BeautifulSoup(r.text, "html5lib")
url_list=[]
a=soup.find('a',class_='btn wide')
page=a['href'].replace('/bbs/HatePolitics/search?
page=','').replace('&q=author%3A'+user_name[0],'')
for i in range(int(page)):
    url_list.append('https://www.ptt.cc/bbs/HatePolitics/search?
page='+str(i+1)+'&q=author%3A'+user_name[0])
title_url=[]
for i in range(int(page)):
    r=requests.get(url_list[i],headers=my_headers)
    soup = BeautifulSoup(r.text, "html5lib")
    b=soup.find_all('div',class_="title")
    for i in range(len(b)):
        if b[i].text.replace('\n','').replace('\t','')[0:8]!='(本文已被刪除)' and
b[i].text.replace('\n','').replace('\t','')[0:3]!='(已被' :
            title_url.append('https://www.ptt.cc'+b[i].a['href'])
```

將所有內文以文字檔做儲存

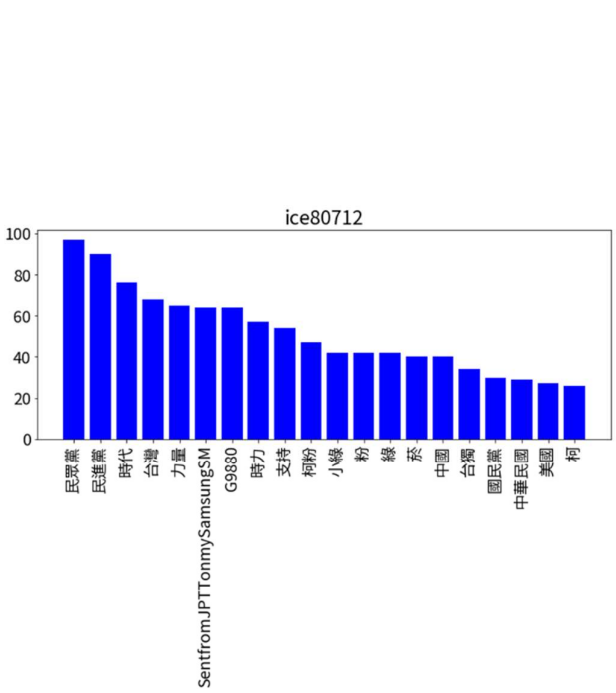
```
for i in range(len(title_url)):
    url = title_url[i]
    r = requests.get(url, headers = my_headers)
    soup = BeautifulSoup(r.text, "html5lib")

    a=soup.find_all('div', class_="bbs-screen bbs-content")
    b=soup.find_all('span',class_='article-meta-value')
    if len(b)==4:
        file = open('ice80712.txt', 'a',encoding="utf-8")
        file.write(str(a[0].contents[4]).replace('\n','').replace(' ',''))
        file.close()
```

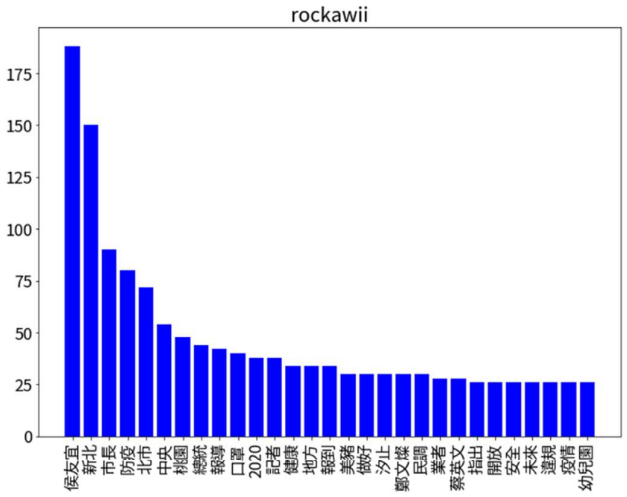
過濾 stopwords

```
text=open('ice80712.txt','r',encoding="utf-8").read()
jieba.set_dictionary('字庫.txt')
with open('停用詞0.txt','r',encoding="utf-8-sig") as f:
    stops=f.read().split('\n')

terms=[]

for t in jieba.cut(text,cut_all=False):
    if t not in stops:
        terms.append(t)
diction=Counter(terms)
```

用長條圖去呈現結果(此處限定為出現次數超過 25 的字詞)

```
dic=diction
keys=[]
values=[]
for key in diction.keys():
    keys.append(key)
for value in diction.values():
    values.append(value)
print(len(keys))
print(len(values))
k=[]
v=[]
for e in dic.values():
    if e>25:
        v.append(e)
for d in dic:
    if dic[str(d)]>25:
        k.append(d)
dic_50=dict(zip(k,v))
kk=[]
vv=[]
for k, v in sorted(dic_50.items(), key=lambda x: -x[1]):
    kk.append(str(k))
    vv.append(v)

import matplotlib.pyplot as plt
plt.rcParams['font.sans-serif'] = ['Taipei Sans TC Beta']
plt.rcParams["font.size"] = 18
colorlist = ["b"]
plt.figure(figsize=(10, 8))
plt.bar(list(kk), vv, color=colorlist,width=0.8)
plt.title('ice80712')
plt.xticks(rotation=90)
plt.tight_layout()

plt.show()
```





7. 最後統計近五個月，每月的發文次數，試著推估是否有在特定期間有明顯的活動特性。

取出帳號內所有發文的日期

```
date=[]

my_headers = {'cookie': 'over18=1;'}
for i in range(int(page)):
    r=requests.get(url_list[i],headers=my_headers)
    soup = BeautifulSoup(r.text, "html5lib")
    b=soup.find_all('div',class_="date")

    for i in range(len(b)):
        date.append(b[i].text.replace(' ',''))
```

統計近五個月每月的發文次數

```
count=[]
for i in range(4,9):
    for d in date:
        if d[0]==str(i):
            count.append(d[0])

mon_count=[]
for i in range(4,9):
    mon_count.append(count.count(str(i)))
```

	月份	文章數
0	4	76
1	5	31
2	6	57
3	7	19
4	8	2

ice80712

	月份	文章數
0	4	25
1	5	16
2	6	8
3	7	0
4	8	7

rockawii

成果說明

(一) 將 ptt 政黑板的高頻率字詞繪製為文字雲圖

文字雲一目了然大家目前所關注的事物。 我選了右手掌這個形狀，象徵的是基督教中的「神的右手」，有著看清一切世局的含意。

(二) 初步的網軍定位

我從五百篇文章中，列出最常出現的幾組 IP，從中選出兩組帳號(ice80712、rockawii)做比較。

上述整理出來的兩張藍色長條圖可以看出，與 ice80712 最相關的三個字詞分別是 民眾檔、民進黨、時代，而 rockawii 則是 侯友宜、新北、市長。明顯看出兩者所關注的不同，但是光靠此數據還是很難去判斷兩者的政治立場。

透過上面整理出的近五個月的發文次數，可看出 4 月到 6 月的次數偏高。可能是受到罷韓事件的影響。

結論

(一)遇到的難題

1.

當初在爬取所有文章內文的時候，有時爬取成功，有時會出現 TypeError。直到後來才發現因為有些文章已被刪除，導致爬取時會出錯，造成程序中斷。因此加上 if 文繞過已刪除的文章就解決問題了。

2.

```
<span class="f2">※超過一行請縮址※</span>
<span class="f2">※請完整轉載原文 請勿修改內文與編排※</span>
<span class="f2">※ 發信站: 批踢踢實業坊(ptt.cc), 來自: 42.75.155.48 (臺灣)
</span>
▼<span class="f2">
  "※ 文章網址: "
  <a href="https://www.ptt.cc/bbs/HatePolitics/M.1599303026.A.DA8.html"
  target="_blank" rel="nofollow">
  https://www.ptt.cc/bbs/HatePolitics/M.1599303026.A.DA8.html</a>
</span>
```

我發現 IP 在 HTML 上是在 class="f2"的地方，但是每一篇文章內都會有無數個重複的 class="f2"，而且數量都不一定，導致無法單一取出 IP。而我的解決方法是先將每一個 class="f2"後面的文字都提取出來，直接利用 re 正規表達函數從文字中過濾出 IP。

(二)檢討 & 改進

1.

目前我是使用同步爬蟲，爬取前 50 頁所有文章大約需要耗時 15 分鐘，因此還有很大的改善空間。 可以嘗試使用 Scrapy 框架或非同步爬蟲來節省爬取時間。

2.

網軍定位其實我還有很多點沒有做到，像是在計算詞頻相似度上，將資料轉換為稀疏向量再用餘弦相似度做計算的方法能夠做到更深入的分析。

(三)心得

現在想想覺得很驚人，從剛開始對網頁爬蟲毫無頭緒，到現在已經可以獨自完成一個專題讓我非常有成就感。 課程共分成了 40 天的學程，但是我足足花了三個月的時間去完成它，我花非常多時間在上網找資料補足自己的不足，雖然遇到了不少挫折，但最終還是熬過去了。

課程給予的學習方向，再加上自學所投入的時間，我在網頁爬蟲以及資料分析的技術上有了重大的突破。而這些技術能夠與日後的自主學習做銜接，對我非常有幫助。