人工智慧

Python斷詞與文字雲教學

jieba, wordcloud套件

吳智鴻

國立臺中教育大學 數位內容科技學系

2019/10/20

斷詞套件

(1) jieba套件 (Python中文斷詞套件)

pip install jieba

安裝繁體中文詞庫

https://raw.githubusercontent.com/fxsjy/jieba/master/extra_dict/dict.txt.big

(2) 文字雲套件 (製做文字雲用)

安裝方式

pip install wordcloud==1.5.0

Jieba 套件

Jieba的特色

支持三種分詞模式：

- 精確模式，試圖將句子最精確地切開，適合文本分析；
- 全模式，把句子中所有的可以成詞的詞語都掃描出來, 速度非常快，但是不能解決歧義；
- 搜索引擎模式，在精確模式的基礎上，對長詞再次切分，提高召回率，適合用於搜索引擎分詞。

支持繁體分詞

支持自定義詞典

Prg#1 Jieba1.ipnb

Jieba的三種斷詞方式

```
#encoding=utf-8
import jieba

seg_list = jieba.cut("我來到台中教育大學",cut_all=True)
print ("Full Mode:", "/ ".join(seg_list)) #全模式

seg_list = jieba.cut("我來到台中教育大學",cut_all=False)
print ("Default Mode:", "/ ".join(seg_list)) #精確模式

seg_list = jieba.cut("我來到台中教育大學") #默認是精確模式
print (", ".join(seg_list))

seg_list = jieba.cut_for_search("志明碩士畢業於台中教育大學，後在日本東京大學深造") #搜索引擎模式
print (", ".join(seg_list))
```

替換為自己想斷詞的句子

```
Full Mode: 我/ 來/ 到/ 台中/ 教育/ 大/ 學
Default Mode: 我來/ 到/ 台/ 中/ 教育/ 大學
我來, 到, 台, 中, 教育, 大學
志明, 碩士, 畢業, 於, 台, 中, 教育, 大學, ，, 後, 在, 日本, 東京大學, 深造
```

Prg#1 Jieba.ipynb

```
sentnce = "我來到台中教育大學就讀碩士"
# 預設模式斷詞
breakword = jieba.cut(sentence, cut_all=False)
print("預設模式:" + '|' . join(breakword))

# 全文模式斷詞
breakword = jieba.cut(sentence, cut_all=True)
print("全文模式:" + '|' . join(breakword))

# 搜尋引擎模式斷詞
breakword = jieba.cut_for_search(sentence)
print("搜尋引擎:" + '|' . join(breakword))
```

```
預設模式:我來|到|台|中|教育|大學
全文模式:我|來|到|台中|教育|大|學
搜尋引擎:我來|到|台|中|教育|大學
```

Jieba的三種斷詞方式

Prg#2 Jieba2.ipynb

下載並加入自己的繁體中文詞典

安裝繁體中文詞庫

https://raw.githubusercontent.com/fxsjy/jieba/master/extra_dict/dict.txt.big

```
#encoding=utf-8
import jieba

# 設定繁體中文詞庫
jieba.set_dictionary('dictionary/dict.txt.big.txt')

sentnce = "我來到台中教育大學就讀碩士"
# 預設模式斷詞
breakword = jieba.cut(sentence, cut_all=False)
print("預設模式:" + '|' . join(breakword))

# 全文模式斷詞
breakword = jieba.cut(sentence, cut_all=True)
print("全文模式:" + '|' . join(breakword))

# 搜尋引擎模式斷詞
breakword = jieba.cut_for_search(sentence)
print("搜尋引擎:" + '|' . join(breakword))
```

```
預設模式:我|來到|台|中|教育|大學
全文模式:我|來到|台中|教育|大學
搜尋引擎:我|來到|台|中|教育|大學
```

沒有斷出台中教育大學

修改中文詞典，讓斷詞更聰明

在繁體中文詞典中加入[台中教育大學]

並重新斷詞

```
584424  龟鹤遐寿 3 i
584425  龟龄鹤算 3 n
584426  龟龙片甲 3 nz
584427  龟龙麟凤 3 ns
584428  龕 5 g
584429  龘 732 zg
584430  台中教育大學 732
```

格式:

所需要的詞　詞頻 詞性(非必要)

例如: 台中教育大學 732 n

詞性: n 名詞, v 動詞

```
#encoding=utf-8
import jieba

# 設定繁體中文詞庫
jieba.set_dictionary('dictionary/dict.txt.big.txt')

sentnce = "我來到台中教育大學就讀碩士"
# 預設模式斷詞
breakword = jieba.cut(sentence, cut_all=False)
print("預設模式:" + '|' . join(breakword))

# 全文模式斷詞
breakword = jieba.cut(sentence, cut_all=True)
print("全文模式:" + '|' . join(breakword))

# 搜尋引擎模式斷詞
breakword = jieba.cut_for_search(sentence)
print("搜尋引擎:" + '|' . join(breakword))
```

```
預設模式:我|來到|台中教育大學
全文模式:我|來到|台中|台中教育大學|教育|大學
搜尋引擎:我|來到|台中|教育|大學|台中教育大學
```

Jieba介紹

TF-IDF

程式碼如下:

```
import jieba.analyse
news = '蘇貞昌表示，春節期間中國武漢肺炎疫情急遽升高，他在年假第一天就到中央流行疫情指揮中心聽取簡報，並宣布提升到二級開設。年假期間，衛福部
tags = jieba.analyse.extract_tags(news, topK=5, withWeight=True)

for tag in tags:
    print('word:', tag[0], 'tf-idf:', tag[1])
```

jieba.analyse.extract_tags 主要有以下的參數:

- sentence 為句子
- topK 代表返回 TF-IDF 權重最大的關鍵字，默認為 20
- withWeight 代表是否返回關鍵字權重值，默認為 False
- allowPOS 代表指定詞性，默認為空，也就是不篩選

輸出如下:

```
word: 疫情 tf-idf: 0.4687853402985507
word: 防疫 tf-idf: 0.4334283486630435
word: 機關 tf-idf: 0.34651500008405794
word: 國人 tf-idf: 0.34651500008405794
word: 感謝 tf-idf: 0.34651500008405794
```

可以看到疫情的最高，實際上句子確實也是疫情出現最多次～符合關鍵字。

資料來源: blog.kennycoder.io/2020/02/12/Python-知名Jieba中文斷詞工具教學/

Jieba介紹

介紹網址

繁體中文版本Jieba

詞性

標籤	含意	標籤	含意	標籤	含意	標籤	含意
n	普通名词	f	方位名词	s	处所名词	t	时间
nr	人名	ns	地名	nt	机构名	nw	作品名
nz	其他专名	v	普通动词	vd	动副词	vn	名动词
a	形容词	ad	副形词	an	名形词	d	副词
m	数量词	q	量词	r	代词	p	介词
c	连词	u	助词	xc	其他虚词	w	标点符号
PER	人名	LOC	地名	ORG	机构名	TIME	时间

資料來源: blog.kennycoder.io/2020/02/12/Python-知名Jieba中文斷詞工具教學/

Jieba介紹

詞性判斷

Jieba 詞性標註功能

透過 jiba.posseg.cut () 可以將句子中的每個斷詞進行詞性標註。

程式碼:

```
words = jieba.posseg.cut('我喜欢写程式')
for word, flag in words:
    print(f'{word} {flag}')
```

輸出如下:

```
我 r
喜欢 v
写 v
程式 n
```

資料來源: blog.kennycoder.io/2020/02/12/Python-知名Jieba中文斷詞工具教學/

Prg#3 Jieba3.ipynb

```python
#encoding=utf-8
import jieba

# 設定繁體中文詞庫
jieba.set_dictionary('dictionary/dict.txt.big.txt')

sentence = "吳智鴻來到台中教育大學數位系就讀碩士"

#jieba.add_word('數位系')
#jieba.add_word('凱特琳')
#jieba.del_word('自定义词')

# 預設模式斷詞
breakword = jieba.cut(sentence, cut_all=False)
print("預設模式:" + '|' . join(breakword))

# 全文模式斷詞
breakword = jieba.cut(sentence, cut_all=True)
print("全文模式:" + '|' . join(breakword))

# 搜尋引擎模式斷詞
breakword = jieba.cut_for_search(sentence)
print("搜尋引擎:" + '|' . join(breakword))
```

可以手動增加/刪除 自定義詞

沒有斷出數位系

```
預設模式:吳智鴻|來到|台中教育大學|數位|系|就讀|碩士
全文模式:吳|智|鴻|來到|台中|台中教育大學|教育|大學|數位|系|就讀|碩士
搜尋引擎:吳智鴻|來到|台中|教育|大學|台中教育大學|數位|系|就讀|碩士
```

Prg#3 Jieba3.ipynb
加入自定義詞 (數位系)

```python
#encoding=utf-8
import jieba

# 設定繁體中文詞庫
jieba.set_dictionary('dictionary/dict.txt.big.txt')

sentence = "吳智鴻來到台中教育大學數位系就讀碩士"

jieba.add_word('數位系')
#jieba.add_word('凱特琳')
#jieba.del_word('自定义词')

# 預設模式斷詞
breakword = jieba.cut(sentence, cut_all=False)
print("預設模式:" + '|' . join(breakword))

# 全文模式斷詞
breakword = jieba.cut(sentence, cut_all=True)
print("全文模式:" + '|' . join(breakword))

# 搜尋引擎模式斷詞
breakword = jieba.cut_for_search(sentence)
print("搜尋引擎:" + '|' . join(breakword))
```

```
預設模式:吳智鴻|來到|台中教育大學|數位系|就讀|碩士
全文模式:吳|智|鴻|來到|台中|台中教育大學|教育|大學|數位|數位系|就讀|碩士
搜尋引擎:吳智鴻|來到|台中|教育|大學|台中教育大學|數位|數位系|就讀|碩士
```

Prg#4 Jieba4.ipynb

自定義詞典

放在 dictionary/user_dict.txt (UTF-8格式)

```
1  數位系
2  國立臺中教育大學
3  國立台中教育大學
4  數位內容科技研究所
5  吳智鴻
```

程式:

```
# 設定繁體中文詞庫
jieba.set_dictionary('dictionary/dict.txt.big.txt')

# 增加自定義停用詞
jieba.load_userdict('dictionary/user_dict.txt')
```

Prg#4 Jieba4.ipynb

```python
#encoding=utf-8
import jieba

# 設定繁體中文詞庫
jieba.set_dictionary('dictionary/dict.txt.big.txt')

# 增加自定義停用詞
jieba.load_userdict('dictionary/user_dict.txt')


sentence = "吳智鴻，來到國立臺中教育大學數位系就讀碩士。"

jieba.add_word('數位系')
#jieba.add_word('凱特琳')
#jieba.del_word('自定义词')

# 預設模式斷詞
breakword = jieba.cut(sentence, cut_all=False)
print("預設模式:" + '|' . join(breakword))

# 全文模式斷詞
breakword = jieba.cut(sentence, cut_all=True)
print("全文模式:" + '|' . join(breakword))

# 搜尋引擎模式斷詞
breakword = jieba.cut_for_search(sentence)
print("搜尋引擎:" + '|' . join(breakword))
```

自定義詞典

User_dict.txt

```
1 數位系
2 國立臺中教育大學
3 國立台中教育大學
4 數位內容科技研究所
5 吳智鴻
```

可以正確斷詞了

```
預設模式:吳智鴻|，|來到|國立臺中教育大學|數位系|就讀|碩士|。
全文模式:吳智鴻|||來到|國立|國立臺中教育大學|臺中|教育|大學|數位|數位系|就讀|碩士||
搜尋引擎:吳智鴻|，|來到|國立|臺中|教育|大學|國立臺中教育大學|數位|數位系|就讀|碩士|。
```

Prg#5 Jieba5.ipynb (增加停用字)

```python
#encoding=utf-8
import jieba

# 設定繁體中文詞庫
jieba.set_dictionary('dictionary/dict.txt.big.txt')

# 增加自定義停用詞
jieba.load_userdict('dictionary/user_dict.txt')

# 打開停用字詞典
with open('dictionary/stopword.txt', 'r', encoding='utf-8-sig') as file:
    stops = file.read().split('\n')  # 將停用詞儲存在stops串列中
print("停用詞:"+'|' . join(stops))

sentence = "吳智鴻，來到國立臺中教育大學數位系就讀碩士。"

#jieba.add_word('數位系')
#jieba.add_word('凱特琳')
#jieba.del_word('自定义词')

# 預設模式斷詞
breakword = jieba.cut(sentence, cut_all=False)
final_words = []   #儲存最後的詞
#拆解句子為字詞
for word in breakword:     # 拆解句子為字詞
    if word not in stops:  #不是停用詞
        final_words.append(word)
print("去除停用:" + '|' . join(final_words))

breakword = jieba.cut(sentence, cut_all=False)
print("預設模式:" + '|' . join(breakword))

# 全文模式斷詞
breakword = jieba.cut(sentence, cut_all=True)
print("全文模式:" + '|' . join(breakword))

# 搜尋引擎模式斷詞
breakword = jieba.cut_for_search(sentence)
print("搜尋引擎:" + '|' . join(breakword))
```

停用字詞典

stopword.txt (UTF8-BOM格式)

```
"
;
,
，
。
來到
```

去除，來到，。

```
停用詞:"|;|,|，|。|來到
去除停用:吳智鴻|國立臺中教育大學|數位系|就讀|碩士
預設模式:吳智鴻|，|來到|國立臺中教育大學|數位系|就讀|碩士|。
全文模式:吳智鴻|||來到|國立|國立臺中教育大學|臺中|教育|大學|數位|數位系|就讀|碩士||
搜尋引擎:吳智鴻|，|來到|國立|臺中|教育|大學|國立臺中教育大學|數位|數位系|就讀|碩士|。
```

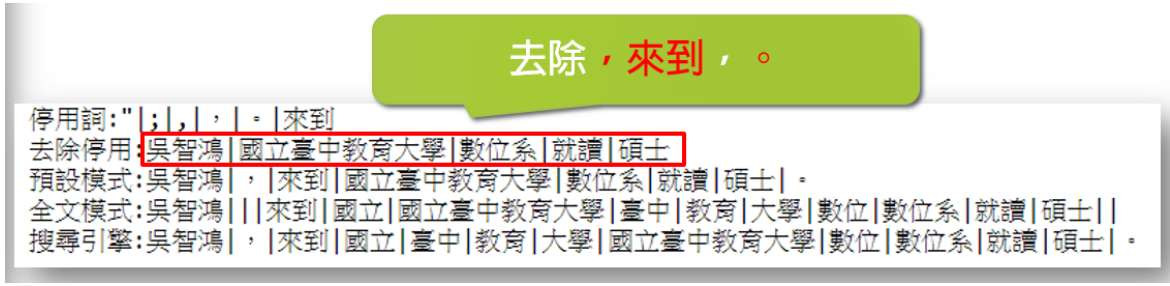
Practice #1 斷詞練習

找一句分析標的文字

1. 可匯入 繁體中文詞典
2. 自定義 使用者辭典 (自己定義幾個需要 優先斷詞的詞)
3. 自定義 停用字　　(自己定義幾個需要停用的詞)

要求:

1. 輸出原先斷詞結果與 修改後的斷詞結果。
2. 張貼程式碼/結果 在FB社團上。
3. 程式碼上傳至你的github



加分題

可以讓Python分析文字情緒。

上網找情緒字典，並設計公式計算每一句話的情緒。

完成者張貼在FB社團上。

須解決問題

- 情緒字典(正/負向辭典)
- 載入情緒辭典並計算正負向情緒詞出現的次數
- 設計並計算情緒公式

進階議題 (自行嘗試看看)

可以讓Python平行斷詞，增加斷詞的效率性。

上網找找有趣的數位人文/教育/等等的文字全文，可供分析。

功能 5)：並行分詞

- 原理：將目標文本按行分隔後，把各行文本分配到多個python進程並行分詞，然後歸併結果，從而獲得分詞速度的可觀提升
- 基於python自帶的multiprocessing模塊，目前暫不支持windows
- 用法：
  - jieba.enable_parallel(4) # 開啓並行分詞模式，參數爲並行進程數
  - jieba.disable_parallel() # 關閉並行分詞模式
- 例子： https://github.com/fxsjy/jieba/blob/master/test/parallel/test_file.py
- 實驗結果：在4核3.4GHz Linux機器上，對金庸全集進行精確分詞，獲得了1MB/s的速度，是單進程版的3.3倍。

WordCloud文字雲套件

文字雲介紹

一個整體形狀很像雲朵的圖形，並且由文字詞頻高低數量所構成

這種由各種字詞組合成、如雲一般的圖形，稱作文字雲(Word Cloud)。我們常在各種社交網站與新聞網站中看到這類圖形的蹤跡，文字雲的存在目的在於能讓閱讀者在不閱讀所有文章的前提下，快速聚焦在大批文章中的主要內容。

http://katiehafner.com/word-cloud/

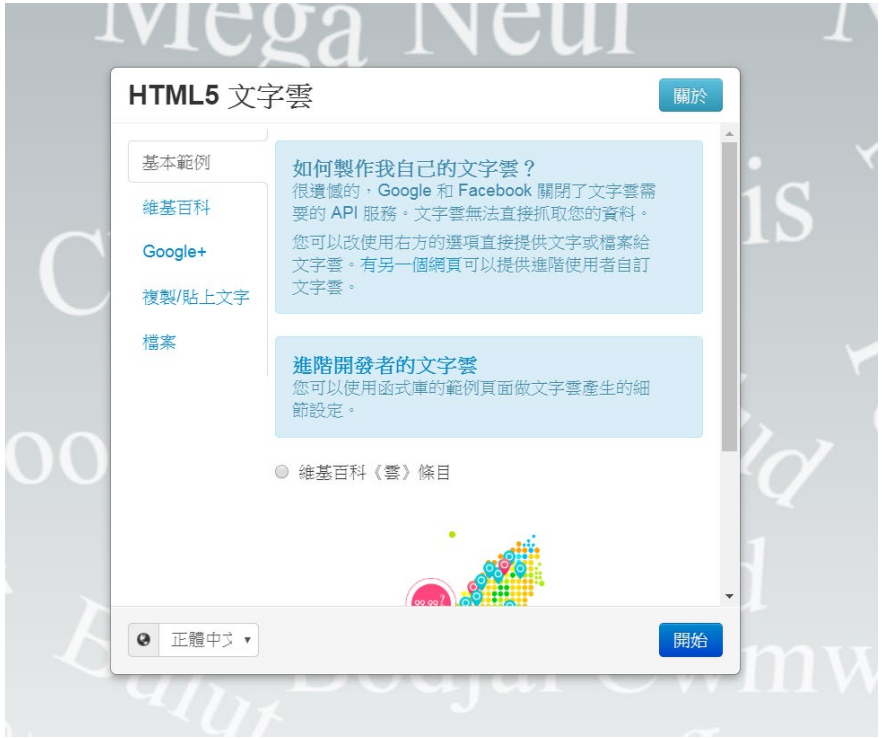
預設形狀的文字雲

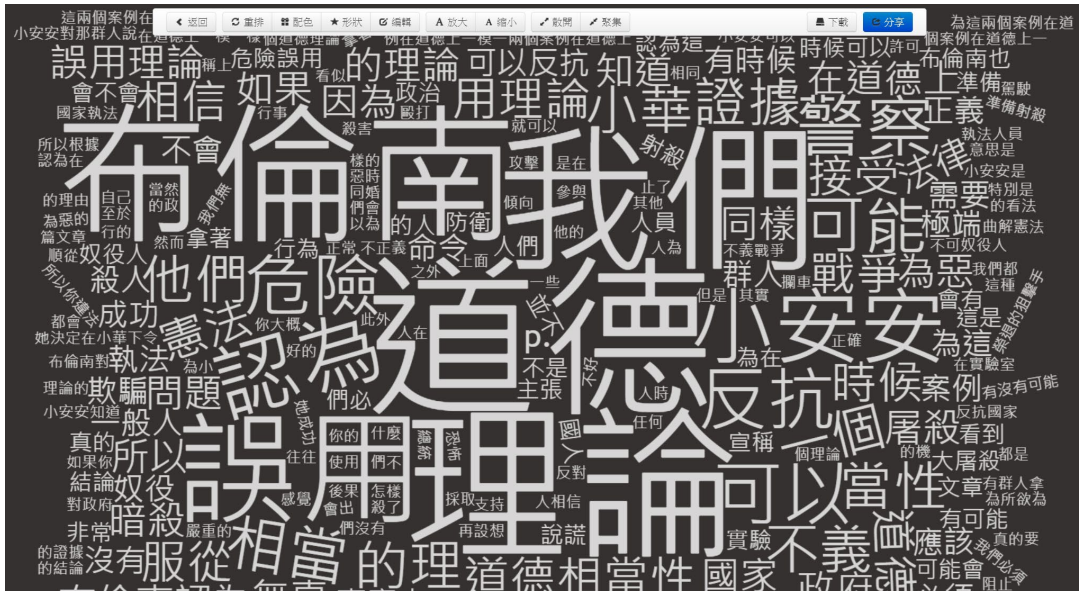
自訂形狀的文字雲

線上玩文字雲

網站上有一個可以玩文字雲的網頁，我們可以把想製作雲朵的文章往上面貼，它就可以畫出漂亮的文字雲，供我們日常分析作使用。

網址如下 : https://wordcloud.timdream.org/#wikipedia:Cloud





文字雲設定

先確認Font 字型名稱

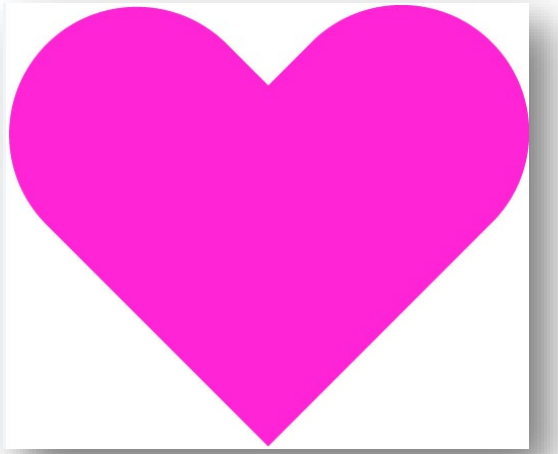
C:\\windows\\font

準備一個讓文字雲分析的文字檔

分析的文字檔 news1.txt

準備 自訂文字雲形狀

heart.png



Prg#6 NewsCloud1.ipynb
自訂文字雲分析

```
from PIL import Image
import matplotlib.pyplot as plt
from wordcloud import WordCloud
import jieba
import numpy as np
from collections import Counter
```

```
text = open('news1.txt', "r",encoding="utf-8").read()  #讀文字資料

jieba.set_dictionary('dictionary/dict.txt.big.txt')
with open('dictionary/stopWord_cloud.txt', 'r', encoding='utf-8-sig') as f:  #設定停用詞
#with open('dictionary/stopWord_cloudmod.txt', 'r', encoding='utf-8-sig') as f:  #設定停用詞
    stops = f.read().split('\n')
terms = []  #儲存字詞
for t in jieba.cut(text, cut_all=False):  #拆解句子為字詞
    if t not in stops:  #不是停用詞
        terms.append(t)
diction = Counter(terms)
# 可列印詞的統計數量
#print(diction)

font = "C:\\Windows\\Fonts\\simsun.ttc"  #設定字型(宋體)
#wordcloud = WordCloud(font_path="C:\\Windows\\Fonts\\simsun.ttc")

mask = np.array(Image.open("heart.png"))  #設定文字雲形狀
#wordcloud = WordCloud(font_path=font)
wordcloud = WordCloud(background_color="white",mask=mask, font_path=font)  #背景顏色預設黑色,改為白色,字體為宋體
wordcloud.generate_from_frequencies(diction)  #產生文字雲

#產生圖片
plt.figure(figsize=(6,6))
plt.imshow(wordcloud)
plt.axis("off")
plt.show()

wordcloud.to_file("news_Wordcloud.png")  #存檔
```

文字雲是根據文字出現的次數多寡來顯示

詞頻


```
Counter({'\n': 56, '」': 33, '「': 33, '道德': 23, '理論': 22, ' ': 19, '布倫南': 17, '誤用': 16, '安安': 11, '認為': 11, '小': 11, '人': 10, '警察': 9, '反抗': 9, '危險': 9, '相當': 8, '一個': 7, '證據': 7, '戰爭': 6, '性': 6, '相信': 6, '接受': 6, '義': 6, '憲法': 6, '小華': 6, '服從': 5, '政府': 5, '中': 5, '法律': 5, '暗殺': 5, '問題': 5, '國家': 5, '同樣': 5, '知道': 5, '有時候': 4, '成功': 4, '案例': 4, '惡': 4, '事實上': 4, '需要': 4, '應該': 4, 'p': 4, '奴役': 4, '無辜': 4, '正義': 4, '殺人': 4, '欺騙': 4, '』': 3, '命令': 3, '準備': 3, '時': 3, '結論': 3, '真的': 3, '『': 3, '主張': 3, '防衛': 3, '看到': 3, '人們': 3, '這是': 3, '信念': 3, '說謊': 3, '宣稱': 3, '射殺': 3, '下令': 2, '反對': 2, '行為': 2, '決定': 2, '選民': 2, '下': 2, '正確': 2, '有人': 2, '一模一樣': 2, '所欲': 2, '嚴重': 2, '設想': 2, '傾向': 2, '駕駛': 2, '群人': 2, '毆打': 2, '執法': 2, '後果': 2, '採取': 2, '實驗室': 2, '支持': 2, '殺害': 2, '攻擊': 2, '違法': 2, '感覺': 2, '看似': 2, '意思': 2, '人為': 2, '使用': 2, '一些': 2, '同婚': 2, '順從': 2, '這種': 2, '提出': 2, '屠殺': 2, '理由': 2, '著槍': 2, '民眾': 2, '文章': 2, '看法': 2, '往往': 2, '執法人員': 2, '政治': 2, '上面': 2, '解讀': 2, '不可': 2, '以為': 2, '兩個': 2, '惡行': 2, '許可': 2, '那群人': 2, '阻止': 2, '大屠
```

Practice #2 文字雲練習

要求

上網找找有趣的數位人文/教育/等等的文字全文，可供分析。

找一個分析標的文字

Jieba斷詞
- 繁體中文詞庫
- 自訂詞庫
- 停用詞

分析詞頻並產生文字雲

設定文字雲成不同樣式
- 不同顏色
- 不同字體 (黑體)
- 不同形狀

輸出

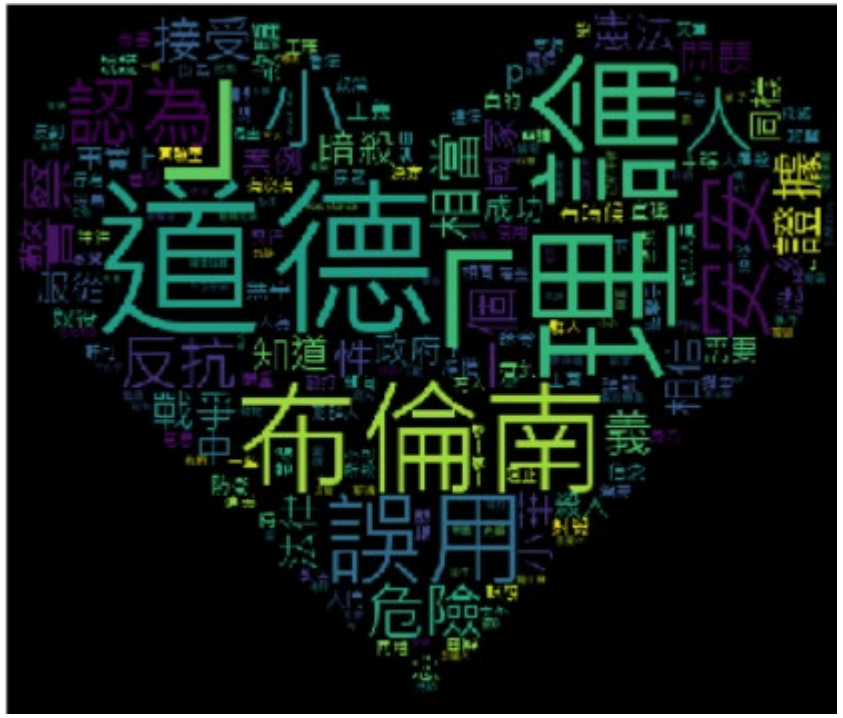
張貼在FB社團上

主題: 例如金庸小說文字雲分析
Why 為何做這主題?

結果: (文字雲圖型)

簡單解釋你發現了甚麼，有甚麼意義

NewsCloud2.ipynb
設定成不同字體與背景



搜尋字體 c:\windows\fonts

需要找出字型的真正英文檔名。

Ex. 微軟正黑體

Prg#7 NewsCloud2.ipynb
自訂文字雲分析　修訂字體與停用字

```
# WordCloud2  文字雲

from PIL import Image
import matplotlib.pyplot as plt
from wordcloud import WordCloud
import jieba
import numpy as np
from collections import Counter
```

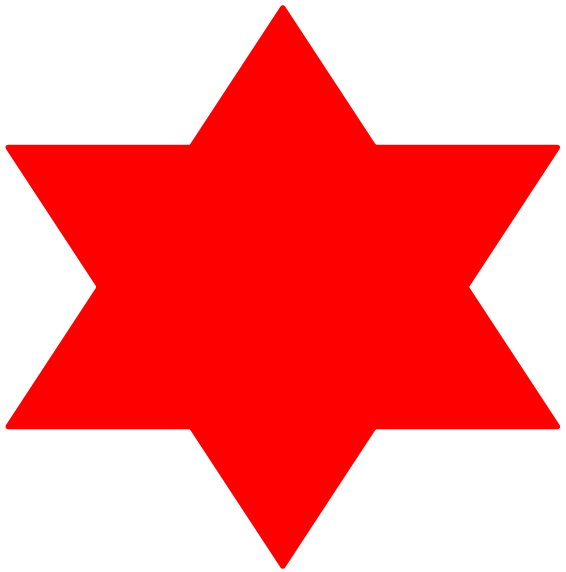
```
text = open('news1.txt', "r",encoding="utf-8").read()  #讀文字資料

jieba.set_dictionary('dictionary/dict.txt.big.txt')
with open('dictionary/stopWord_cloud_new.txt', 'r', encoding='utf-8-sig') as f:  #設定停用詞
#with open('dictionary/stopWord_cloudmod.txt', 'r', encoding='utf-8-sig') as f:  #設定停用詞
    stops = f.read().split('\n')
terms = []  #儲存字詞
for t in jieba.cut(text, cut_all=False):  #拆解句子為字詞
    if t not in stops:  #不是停用詞
        terms.append(t)
diction = Counter(terms)
# 可列印詞的統計數量
#print(diction)

#font = "C:\\Windows\\Fonts\\simsun.ttc"  #設定字型(宋體)
font = "C:\\Windows\\Fonts\\MSJH.ttf"  #設定字型(宋體)
#wordcloud = WordCloud(font_path="C:\\Windows\\Fonts\\simsun.ttc")

mask = np.array(Image.open("star.png"))  #設定文字雲形狀
#wordcloud = WordCloud(font_path=font)
wordcloud = WordCloud(background_color="white",mask=mask, font_path=font)  #背景顏色預設黑色,改為白色,字體為宋體
wordcloud.generate_from_frequencies(diction)  #產生文字雲

#產生圖片
plt.figure(figsize=(6,6))
plt.imshow(wordcloud)
plt.axis("off")
plt.show()

wordcloud.to_file("news_Wordcloud.png")  #存檔
```

原先的圖案

自訂的圖案
並去除「」符號

Heart.png

start.png



Practice #3 爬蟲結合文字雲練習

要求

與先前爬蟲程式結合。

可輸入一個關鍵字，然後至google搜尋該關鍵字所有文章標題。然後將標題文字給Jieba斷詞，並產生文字雲分析趨勢。

找一個分析標的文字

Jieba斷詞

- 繁體中文詞庫
- 自訂詞庫
- 停用詞

分析詞頻並產生文字雲

輸出

張貼程式碼/輸出結果在FB社團上

HINT

串接部分

1. 把Google搜尋到的標題（i.text)，組合成為一個大字串(text)。
2. 然後再把text這個含有所有標題的大字串給Jieba斷詞

```
#讀取文字檔資料
#text = open('news1.txt', "r",encoding="utf-8").read()  #讀文字資料

#讀取Google資料

text = ''
for i in items:
    # 標題
    text = text + i.text
print( text)
```

想辦法串接這兩個（解答）

原先Google搜尋程式

```
import requests
from bs4 import BeautifulSoup

# Google 搜尋 URL
google_url = 'https://www.google.com.tw/search'

# 查詢參數
my_params = {'q': '寒流'}

# 下載 Google 搜尋結果
r = requests.get(google_url, params = my_params)

# 確認是否下載成功
if r.status_code == requests.codes.ok:
  # 以 BeautifulSoup 解析 HTML 原始碼
  soup = BeautifulSoup(r.text, 'html.parser')

  # 觀察 HTML 原始碼
  #print(soup.prettify())

  # 以 CSS 的選擇器來抓取 Google 的搜尋結果
  items = soup.select('div.kCrYT > a[href^="/url"]')

  for i in items:
    # 標題
    print("標題：" + i.text)
    # 網址
    print("網址：" + i.get('href'))
```

斷詞與文字雲程式

```
from PIL import Image
import matplotlib.pyplot as plt
from wordcloud import WordCloud
import jieba
import numpy as np
from collections import Counter

#讀取文字檔資料
#text = open('news1.txt', "r",encoding="utf-8").read()  #讀文字資料

#讀取Google資料

text = ''
for i in items:
    # 標題
    text = text + i.text
print( text)

jieba.set_dictionary('dictionary/dict.txt.big.txt')
with open('dictionary/stopWord_cloud.txt', 'r', encoding='utf-8-sig') as f:  #設定停用詞
#with open('dictionary/stopWord_cloudmod.txt', 'r', encoding='utf-8-sig') as f:  #設定停用詞
    stops = f.read().split('\n')
terms = []  #儲存字詞
for t in jieba.cut(text, cut_all=False):  #拆解句子為字詞
    if t not in stops:  #不是停用詞
        terms.append(t)
diction = Counter(terms)
# 可列印詞的統計數量
#print(diction)

#font = "C:\\Windows\\Fonts\\simsun.ttc"  #設定字型(宋體)
font = "C:\\Windows\\Fonts\\MSJH.ttf"  #設定字型(宋體)
#wordcloud = WordCloud(font_path="C:\\Windows\\Fonts\\simsun.ttc")

mask = np.array(Image.open("heart.png"))  #設定文字雲形狀
#wordcloud = WordCloud(font_path=font)
wordcloud = WordCloud(background_color="white",mask=mask, font_path=font)  #背景顏色預設黑色,改為白色,字體為宋體
wordcloud.generate_from_frequencies(diction)  #產生文字雲

#產生圖片
plt.figure(figsize=(6,6))
plt.imshow(wordcloud)
plt.axis("off")
plt.show()

wordcloud.to_file("news_Wordcloud.png")  #存檔
```

NLP 自然語言處理程式

情感分析、自動文章摘要

NLP可以做到什麼

自動問答（Question Answering，QA）：

- 它是一套可以理解複雜問題，並以充分的準確度、可信度和速度給出答案的計算系統，以IBM‘s Waston為代表；

資訊抽取（Information Extraction，IE）：

- 其目的是將非結構化或半結構化的自然語言描述文本轉化結構化的資料，如自動根據郵件內容生成Calendar；

情感分析（Sentiment Analysis，SA）：

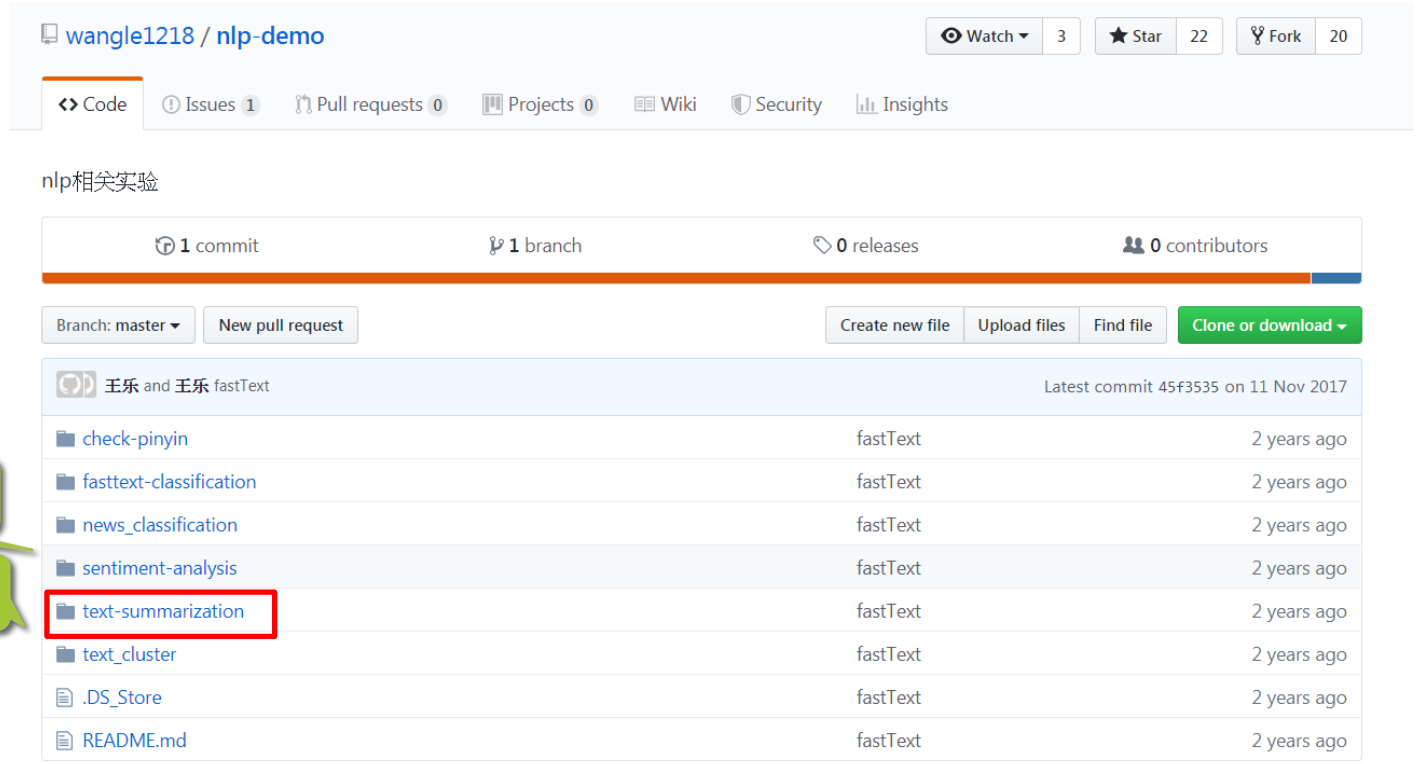
- 又稱傾向性分析和意見挖掘，它是對帶有情感色彩的主觀性文本進行分析、處理、歸納和推理的過程，如從大量網頁文本中分析使用者對“數碼相機”的“變焦、價格、大小、重量、閃光、易用性”等屬性的情感傾向；

機器翻譯（Machine Translation，MT）：

- 將文本從一種語言轉成另一種語言，如中英機器翻譯。

NLP的一個Github
裡面有幾個NLP的範例程式

https://github.com/wangle1218/nlp-demo



情感分析

自動摘要

自動摘要

自動摘要

可以用原始文章內部的部分句子來代表整個文章。

自動摘要可以把大量文字，濃縮成較少的摘要，節省大量閱讀資訊的時間。

需要套件

把autosummary.py 放在python程式資料夾，再載入使用。

使用方式：

```
import AutoSummary as ausu
```

該程式係參考以下程式修改而成

https://github.com/wangle1218/nlp-demo/tree/master/text-summarization

Prg#7. summary1.ipynb
讀取文字檔自動摘要

```
import AutoSummary as ausu

content = 'issue1.txt'
with open(content, 'r', encoding='utf8') as f:  #讀取原始文章
    text = f.read()

stops = []
with open('dictionary/stopWord_summar.txt','r', encoding='utf8') as f:  #停用詞庫
    for line in f.readlines():
        stops.append(line.strip())

sentences,indexs = ausu.split_sentence(text)  #按標點分割句子
tfidf = ausu.get_tfidf_matrix(sentences,stops)  #移除停用詞並轉換為矩陣
word_weight = ausu.get_sentence_with_words_weight(tfidf)  #計算句子關鍵詞權重
posi_weight = ausu.get_sentence_with_position_weight(sentences)  #計算位置權重
scores = ausu.get_similarity_weight(tfidf)  #計算相似度權重
sort_weight = ausu.ranking_base_on_weigth(word_weight, posi_weight, scores, feature_weight = [1,1,1])  #按句子權重排序
summar = ausu.get_summarization(indexs,sort_weight,topK_ratio = 0.1)  #取得摘要
print('原文:\n', text)
print('============================================================')
print('摘要:\n',summar)
```

```
============================================================
摘要:
 台南出生的黃仁勳，25 年前在美國創辦的 NVIDIA，近年來屢創高峰，股價成長超過 700%、營收數字成長 94%。」

台灣時間雙十國慶的傍晚， 在德國慕尼黑，NVIDIA（輝達）執行長黃仁勳（JensenHuang）， 正準備上台發表 2018 第三場的全球 GPU 技術大會（GPU TechnologyConference，簡稱 GTC）。

黃仁勳也談到醫療的應用，尤其是在許多生物影像的判讀上，人工智慧可以讓醫師的工作變得更省力卻更精準；最後，他強調虛擬實境（VR）已跳脫遊戲的範疇，營建業可用來創造建築實景，車廠也可以拿來設計超跑。
```

Prg#8 summary_union.ipynb
讀取新聞網站新聞

```python
import AutoSummary as ausu
import requests
from bs4 import BeautifulSoup as soup

stops = []
with open('dictionary/stopWord_summar.txt','r', encoding='utf8') as f:  #停用詞庫
    for line in f.readlines():
        stops.append(line.strip())

urls = []
url = 'https://udn.com/news/breaknews/1'  #聯合報新聞
html = requests.get(url)
sp = soup(html.text, 'html.parser')
data1 = sp.select('#breaknews_body dl dt h2 a')
for d in data1:  #取得新聞連結
    urls.append('https://udn.com' + d.get('href'))

i = 1
for url in urls:  #逐一取得新聞
    html = requests.get(url)
    sp = soup(html.text, 'html.parser')
    data1 = sp.select('#story_body_content p')  #新聞內容
    print('處理第 {} 則新聞'.format(i))
    text = ''
    for d in data1:
        if d.text.find('延伸閱讀') != -1:  #遇到延伸閱讀就結束此則新聞
            break
        if d.text != '':  #有新聞內容
            text += d.text
    sentences,indexs = ausu.split_sentence(text)  #按標點分割句子
    tfidf = ausu.get_tfidf_matrix(sentences,stops)  #移除停用詞並轉換為矩陣
    word_weight = ausu.get_sentence_with_words_weight(tfidf)  #計算句子關鍵詞權重
    posi_weight = ausu.get_sentence_with_position_weight(sentences)  #計算位置權重
    scores = ausu.get_similarity_weight(tfidf)  #計算相似度權重
    sort_weight = ausu.ranking_base_on_weigth(word_weight, posi_weight, scores, feature_weight = [1,1,1])
    summar = ausu.get_summarization(indexs,sort_weight,topK_ratio = 0.3)  #取得摘要
    print(summar)
    print('============================================================')
    i += 1
```

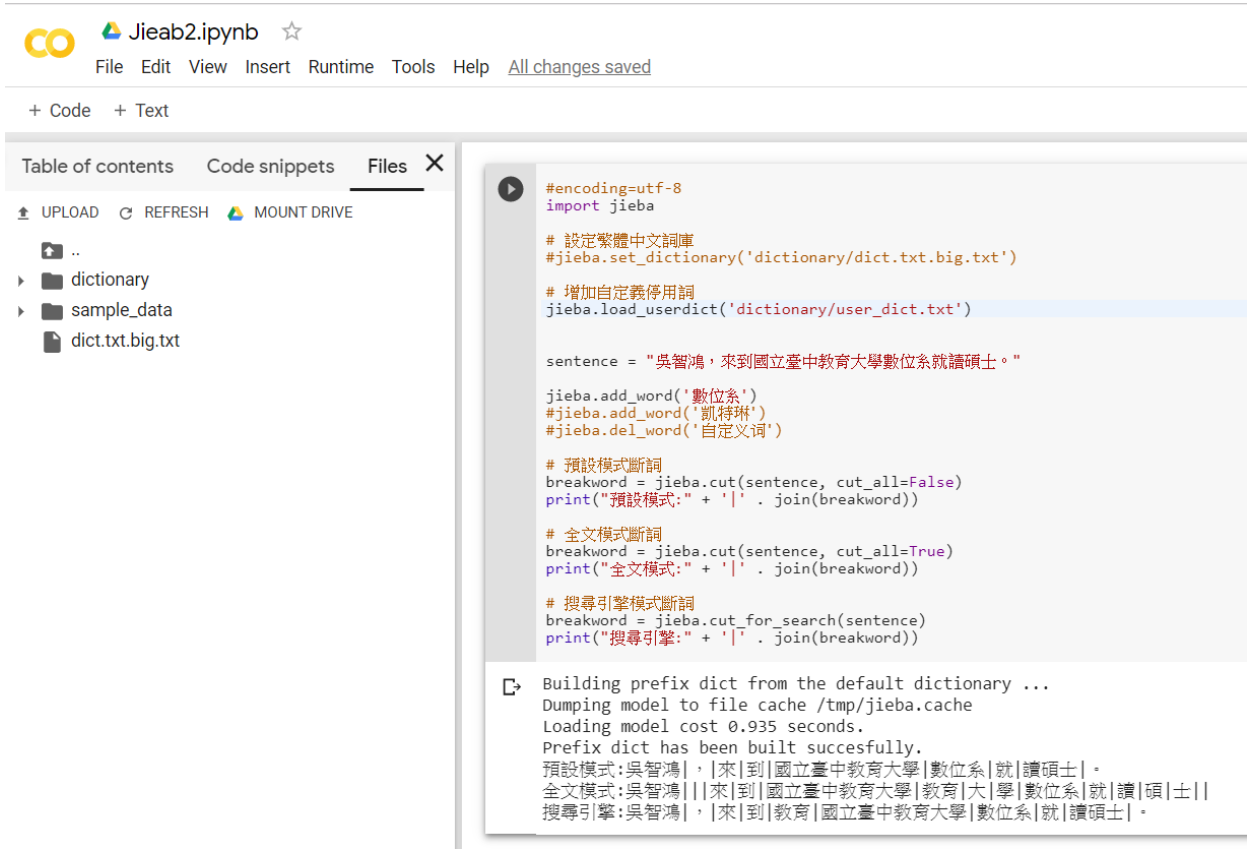
需針對不同網頁設計去調整

需針對不同網頁設計去調整

```
處理第 1 則新聞
Loading model cost 0.778 seconds.
Prefix dict has been built succesfully.
保險局預計11月將公布明年壽險業各幣別新契約責任準備金利率，但今天財委會上調降幅度提前曝光，預計新台幣、美元、人民幣保單責任準備金利率將調降1碼，10年期以上澳幣保單將調降2碼。顧立雄表示，保險公司每年保費增加幅度都在2兆多，現在之所以調降責任準備金利率，是因為各幣別公債市場殖利率都較去年下降，因此責任準備金利率在新台幣跟美元都較去年調降一碼，大家也都同意要調降。2019年10月21日 - 2019年11月20日活動期間您可獲得活動金幣 3,000 枚，當您看到優質新聞，即可點按文章中的「贊助好新聞」按鈕贊助該篇文章，且可隨時至會員中心查詢目前金幣的使用狀況。越多天登入贊助，中獎機率越高點按文章中的「贊助好新聞」，以活動金幣贊助該篇文章，支持心中優質新聞。據保險局的規劃，明年新台幣、美元、人民幣保單責任準備金都將調降1碼、10年期以上的澳幣保單預計調降最多達2碼。
============================================================
處理第 2 則新聞
10月是國際反霸凌月，教育部最近在官方臉書粉絲專頁秀出八件中學制服，上面所繡的名字包括「娘炮」、「太平洋」、「高額頭」、「自以為混血」、「那個沒屌的」等，引起網友熱議。教育部也邀八名知名網紅合作，請他們分享過去被霸凌的經驗，今天將在官方臉書上釋出。2019年10月21日 - 2019年11月20日活動期間您可獲得活動金幣 3,000 枚，當您看到優質新聞，即可點按文章中的「贊助好新聞」按鈕贊助該篇文章，且可隨時至會員中心查詢目前金幣的使用狀況。越多天登入贊助，中獎機率越高點按文章中的「贊助好新聞」，以活動金幣贊助該篇文章，支持心中優質新聞。
```

程式來源：文淵閣工作室，「Python機器學習與深度學習特訓班：看得懂也會做的AI人工智慧實戰」，碁峰出版社

Google Colab



參考來源

文淵閣工作室，「Python機器學習與深度學習特訓班：看得懂也會做的AI人工智慧實戰」，碁峰出版社。

網頁資料

https://www.twblogs.net/a/5b7ca46f2b71770a43dbf2ce