作品名稱：文字雲新聞與情分析

一、說明

本專題主要使用 wordcloud 透過 Python、R 程式及 SQLServer 軟體及搭配新聞文字進行字詞拆分繪製出文字雲圖，先在新聞網站上下載文字然後儲存於記事本上。使用 Python 執行文字雲前結果可以搭配 png 圖片產出文字雲效果、使用 R 執行文字雲前結果可以打上數字、英文或符號產出文字雲效果。

二、論文

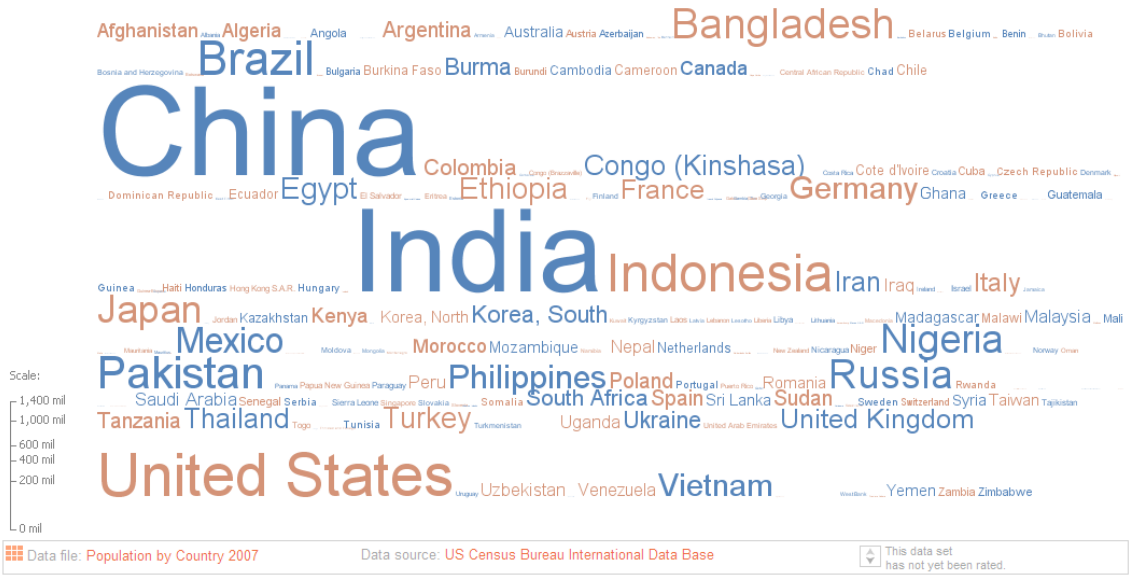
根據文字雲的作用(而非樣式)，在應用中可以將其分成三大類。其中，第一類用於描述網站中的每個獨立條目，而第二類則著力於從整體上刻畫網站所有條目的標籤情況：

第一類文字雲：每一個條目都有自己獨立的文字雲，標籤字體越大，此條目中用戶使用過這個標籤的次數就越多，在頁面公開統計點擊且不要求精準數據的情況下十分適用。如 Last.fm 就是使用了第一類文字雲。

第二類文字雲：網站一般會有一個超大型文字雲，標籤字體越大，網站裡使用過這個標籤的條目數就越多。第二類文字雲可以顯示出標籤的熱門程度，在實際應用中更為常見，如 flickr。

第三類文字雲：在此類中，標籤作為一個數據項目的工具，用於表示在整個集合中里各個項目的數據量的大小。

從廣義來說，相似的可視化技術並不限於用於文字雲，例如還可以用在文字雲或數據雲上。



大多數文字雲通過內嵌 HTML 元素的方式實現。標籤可以按字母次序、重要次序、隨機次序等方式排列。部分網站還應用語義分組技術，讓內容相聯的標籤聚攏在一起。而啟發法更可以用於壓縮文字雲的體積，幫助標籤分組。

三、實作

文字雲是關鍵詞的視覺化呈現，將各種關鍵詞的重要性透過字體大小及顏色來表現讓觀看者一目了然。文字雲的形狀可以任意設定，更能添加文字雲的千變萬化

的魅力。首先將要繪製成文字雲圖的新聞文字複製在記事本上面，使用 jieba 套件及 wordcloud 後就可以將繪製好的文字雲圖產生文字效果，每執行一次就會有不同的變化。

2020Q1房市走向 57%消費者擔心餘屋賣壓
2019年房市明顯回溫，根據永慶房產集團調查顯示，針對2020年第1季房市走向，全台有57%消費者擔心餘屋帶來房市賣壓較上一季略減，但依舊有32%。
永慶調查結果顯示，全台共57%消費者擔心大量餘屋帶來房市賣壓，擔憂情緒最高的落在高雄市，占比達到68%，其次是桃園
不過，在房市低利、高成數的購屋條件下，調查結果顯示，消費者依舊認為房地產是資金避險的首選，雖然占比略降3個百分
永慶房產集團業務總經理葉凌棋表示，2020年國內經濟持續緩步復甦，同時有台商回流、轉單效應發酵，今年經濟前景值得
定去化，成為2020年房市交易值得觀察重點。

一、 R 語言繪製文字雲：

使用 Rsutdio 軟體的 wordcloud 及 wordcloud2 的套件繪製成文字雲圖，然後另存為圖片。ch_news = c("請匯入新聞文字")需要再網站或記事本內找相關新聞文字複製並貼在"請匯入新聞文字"上面。

程式碼:

```
library(jiebaR)
library(wordcloud) # 第一代文字雲
library(wordcloud2) # 第二代文字雲
library(RColorBrewer)

ch_news = c("請匯入新聞文字") # 匯入新聞

new_terms = c("鄉鎮縣市", "民意代表", "店家", "買賣交屋")
writeLines(new_terms, '/Users/hello/Documents/R/ch_terms.txt') # 自訂字典,請輸入自己的路徑

stopwords = c( "在","的","上", "下","是", "個","來","為","亦","或", "之", "與", "於", "用", "都", "等", "日", "月", "年", "週", "嗎", "以", "就", "但", "及", "也", "了", "要", "不", "會", "和", "對", "著", "後", "她", "他")
writeLines(stopwords, '/Users/hello/Documents/R/ch_stopwords.txt') # 自訂停用詞,請輸入自己的路徑

cutter = worker(user='/Users/hello/Documents/R/ch_terms.txt', stop_word = '/Users/hello/Documents/R/ch_stopwords.txt') # 引用字典和停用詞,請輸入自己的路徑

ch_news <- gsub("[0-9a-zA-Z]+?", "", ch_news) # 刪除數字和字母

ch_news <- cutter[ch_news] # 斷詞
```

```
freq_ch <- sort(table(ch_news), T)
freq_ch = as.data.frame(freq_ch)
colnames(freq_ch) <- c("Words", "Freq") # 字頻表


head(freq_ch) # 查看前 10 筆資料


par(family=("NotoSansCJKtc-Medium")) # 設定字體 Mac

customed_colors = c("#000080", "#ffff00", "#6495ed", "#00bfff",
"#87cefa", "#db7093", "#ba55d3", "#b22222", "#008080", "#ff8c00",
"#6b8e23") # 顏色

ch_wordcloud = wordcloud(freq_ch$Words, freq_ch$Freq, min.freq = 2,
random.order = F, ordered.colors = F, colors = customed_colors);
ch_wordcloud # 第一代文字雲

ch_wordcloud2 = wordcloud2(freq_ch, size = 1.3, color =
customed_colors, backgroundColor="white"); ch_wordcloud2
# 第二代文字雲

letterCloud(freq_ch,'OK',size = 0.50)
letterCloud(freq_ch,'R',size = 0.50)
```

執行結果:

繪製好的文字雲圖將存成圖檔:請點選 Export/Save as an image file/請選好路徑及命名/save 即可。(執行文字符號時出現文字雲結果反應很慢甚至沒出現文字雲需要多式幾次就會成功)

二、 Python 繪製文字雲：

使用 Python 軟體的 wordcloud 的套件繪製成文字雲圖，執行後會自動另存為圖片。

程式碼:

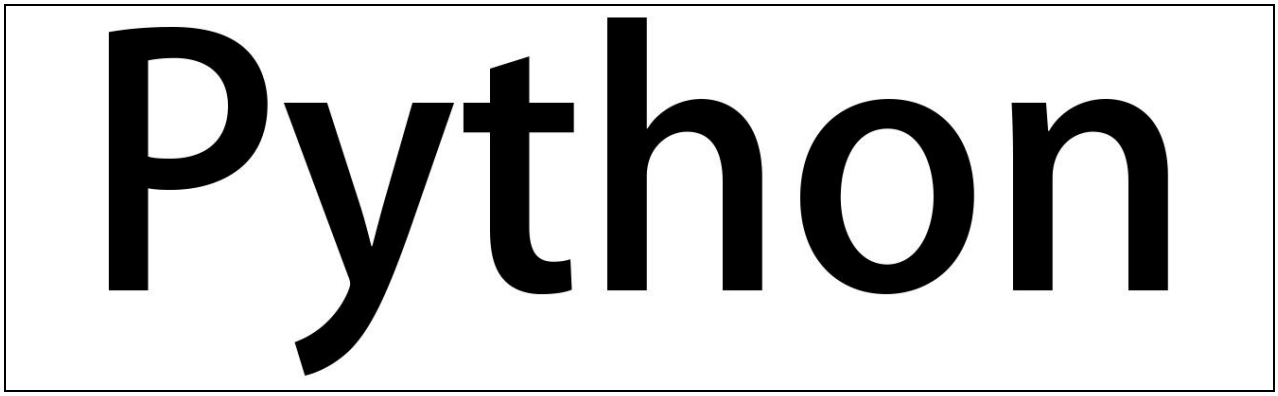
```
#程式
%matplotlib inline
import matplotlib.pyplot as plt
from wordcloud import WordCloud
from PIL import Image
import jieba
import numpy as np
# 如果檔案內有一些編碼錯誤，使用 errors='ignore' 來忽略錯誤
with open("財經_房地產_1.txt", encoding="Utf-8", errors='ignore') as f:
    text = f.read()

# 設定使用 big5 斷詞
jieba.set_dictionary('./dict.txt.big')
wordlist = jieba.cut(text)
words = " ".join(wordlist)
#文字雲造型圖片
mask = np.array(Image.open('pythonM 拷貝.jpg')) #文字雲形狀
# 從 Google 下載的中文字型
font = 'SourceHanSansTW-Regular.otf'
#背景顏色預設黑色，改為白色、使用指定圖形、使用指定字體
my_wordcloud =
```

```
WordCloud(background_color='white',mask=mask,font_path=font).generate(words)

plt.imshow(my_wordcloud)
plt.axis("off")
plt.show()
#存檔
my_wordcloud.to_file('pythonword_cloud.png')
```

執行結果前:



執行結果後:

背景一定要是純白色的才會配合黑字或黑圖演算法走，請不要用透明背景只有黑字或黑圖來執行出文字雲，這樣子會把整個文字圖給籠罩起來(天上少有，地下難尋)。

三、 SQLServer 繪製文字雲：

首先使用 WinLoadNews 上傳器將新聞文字 txt 上傳至 SSMS 資料庫內接者進行步驟執行 1 檢視新聞文字資料、2.指定新聞文字號碼、3.斷詞分析、4.斷詞分類結果、5.讀取資料、6.建立資料、7.宣告變數、8.文字斷詞分類結果、最後繪製文字雲圖。

程式碼:

使用 SQL Server Management Studio(SSMS)內建 Python 套件(jieba、wordcloud 套件)，使用 R 語言及 Python 繪製成文字雲圖，執行後會自動另存為圖片。

```
WITH Temp
```

```
AS
(
    SELECT [NewsId] FROM [News] WHERE SUBSTRING([Label],1,1)='1'
) #請確定好新聞 ID
,Temp2
AS
(
    SELECT A.[NewsId],B.KeyWords,B.Cnts
    FROM [Temp] AS A CROSS APPLY(SELECT * FROM
dbo.GetKeyWords(A.NewsId)) AS B
)
SELECT TOP(300) [KeyWords],SUM([Cnts]) AS Cnts
FROM Temp2
GROUP BY [KeyWords]
ORDER BY [Cnts] DESC

-----------------------------------------

EXEC sp_execute_external_script
    @language = N'Python'
    , @script = N'
import matplotlib.pyplot as plt
from wordcloud import WordCloud
import pandas as pd

input_data = InputDataSet
df=pd.DataFrame(input_data)
nn = list(df.KeyWords)
vv = list(df.Cnts)
my_dict=dict(zip(nn,vv))

#設定中文字體
font_path = "C:\Windows\Fonts\kaiu.ttf"

wc = WordCloud(font_path = font_path,width = 800, height =
800,background_color ="white",stopwords=None,min_font_size = 10)
wc.generate_from_frequencies(my_dict)
wc.to_file("C:\\DD\\p3.png")
```

```
'
, @input_data_1 = N'
WITH Temp
AS
(
    SELECT [NewsId] FROM [News] WHERE SUBSTRING([Label],1,1)=''1''
) #請確定好新聞 ID
,Temp2
AS
(
    SELECT A.[NewsId],B.KeyWords,B.Cnts
    FROM [Temp] AS A CROSS APPLY(SELECT * FROM
dbo.GetKeyWords(A.NewsId)) AS B
)
SELECT TOP(300) [KeyWords],SUM([Cnts]) AS Cnts
FROM Temp2
GROUP BY [KeyWords]
ORDER BY [Cnts] DESC
'
```

執行結果:

繪製好的文字雲圖目前只有單調的形狀，畢竟它也是結果也已經很不錯了。

四、結論

使用 WordCloud 繪製出文字雲前 1.R 語言要重頭開始執行到結尾將複製好的新聞文字貼在指定的位置然後就可以繪製成文字雲圖。它是用文字、數字、標點符號繪製成文字雲圖形狀。2.Python 比較單純；將圖片、記事本等要放同一個資料夾及相同路徑就直接執行一整個程式碼就可以繪製成文字雲圖。用純白背景及黑字或黑圖繪製成文字雲圖形狀。3.SQLServer 必須要跟 Python、R 關聯且下載執行套件，然後開啟 SSMS 軟體建立新聞資料庫，先上傳新聞資料再清洗新聞資料及檢查，最後繪製成文字雲圖。目前只有單調的形狀，畢竟它也是結果也已經很不錯了。
文字雲是可視化的文檔用詞頻率統計權重表，這項技術最近常用於具體化、形象化政治演講的話題和內容。

五、參考

好學校 Hahow R 語言系列課程。
資展國際 R 語言系列課程。
巨匠電腦 Azure for ML 及 R 語言資料分析。