Lonely Caesar

文字雲新聞與情分析

作者　吳彥瑾

作品名稱：文字雲新聞與情分析

一、說明

本專題要使用（Word Cloud）製作文字雲並進行新聞情感分析是一個非常有趣且實用的專題。以下是使用 Python 和 R 來連接 SQL Server 並製作文字雲的詳細步驟說明。

首先，需要確保您的開發環境已經安裝好必要的軟件和庫：

1. Python：安裝 Anaconda3.8（推薦）或直接安裝 Python。
2. R：安裝 R4.3.2 和 RStudio。
3. SQL Server：確保已經安裝並運行 SQL Server2019，並且有權限訪問相關數據庫。
4. 將抓複製下來的新聞文字貼在記事本.txt 上面然後存檔，存檔時編碼請選 utf-8，這樣就不會出現亂碼。
5. 需要有 dict.txt.big.txt、中文字型就能夠執行出成果。

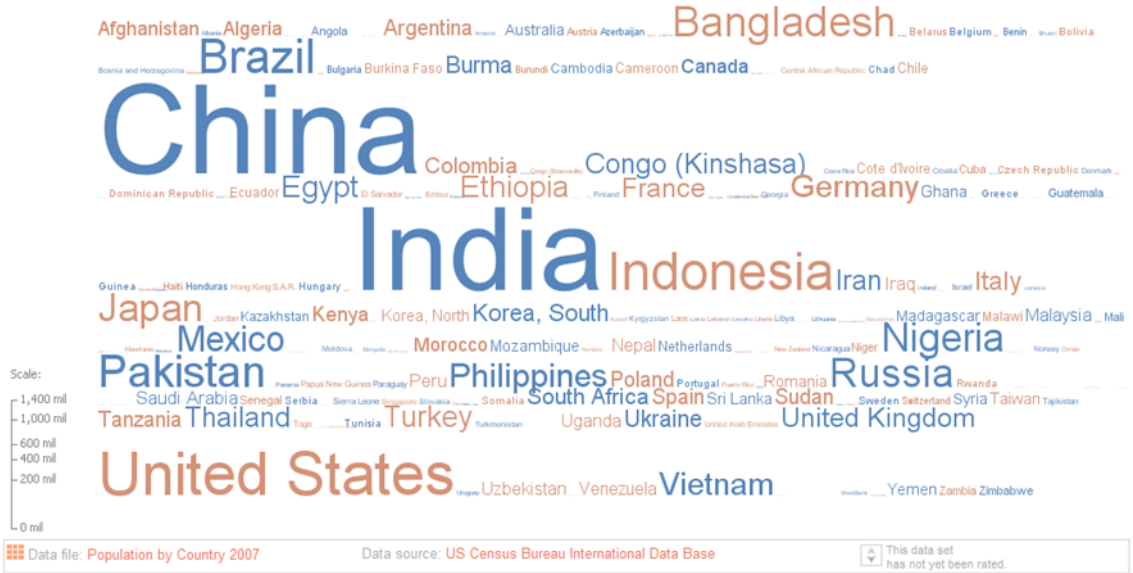
二、相關文章

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。

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。

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。

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。

三、實作

針對已經貼好的新聞文字及命名的.txt 作執行。

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、 使用 R 語言繪製文字雲：

使用 R 軟體文字斷詞、分詞及繪製成文字雲圖，執行手動另存為圖片。

輸入 install.packages("jiebaR")指令執行套件。

輸入 install.packages("wordcloud")指令執行套件。

輸入 install.packages("wordcloud2")指令執行套件。

輸入 install.packages("RColorBrewer")指令執行套件。

程式碼:

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

# 設定字體
par(family="SourceHanSansTW-Regular")

# 匯入新聞文本
ch_news <- readLines("財經_房地產_1.txt", encoding = "UTF-8")

# 定義字典、斷詞和停用詞
cutter <- worker(user='dict.txt.big.txt', stop_word =
'dict.txt.big.txt')
ch_news <- gsub("[0-9a-zA-Z]+?", "", ch_news) # 刪除數字和字母
ch_news <- cutter[ch_news] # 斷詞

# 計算字頻
freq_ch <- sort(table(ch_news), decreasing = TRUE)
freq_ch <- as.data.frame(freq_ch)
colnames(freq_ch) <- c("Words", "Freq") # 字頻表

# 設定自訂顏色
customed_colors <- c("#000080", "#ffff00", "#6495ed", "#00bfff",
"#87cefa", "#db7093", "#ba55d3", "#b22222", "#008080", "#ff8c00",
"#6b8e23")
```

```
# 生成第一代文字雲
ch_wordcloud <- wordcloud(freq_ch$Words, freq_ch$Freq, min.freq = 2,
random.order = FALSE, ordered.colors = FALSE, colors = customed_colors)
ch_wordcloud
# 生成第二代文字雲
ch_wordcloud2 <- wordcloud2(freq_ch, size = 1, color = customed_colors,
backgroundColor="white")
ch_wordcloud2
# 使用 letterCloud 函數生成字母文字雲
letterCloud(freq_ch, 'OK', size = 0.5)
letterCloud(freq_ch, 'R', size = 0.5)
wordcloud2(freq_ch, size = 0.4,shape = 'star')
```

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)

註：從 2023 年 11 月底使用 letterCloud 函數無法將文字運行出造型。

解決方式：

```
options("repos" = c(CRAN="https://mirrors.tuna.tsinghua.edu.cn/CRAN/"))
install.packages("jsonlite")
library('devtools')
devtools::install_github("lchiffon/wordcloud2")
```

二、 使用 Python 繪製文字雲：

使用 Python 軟體文字斷詞、分詞及繪製成文字雲圖，執行後會自動另存為圖片。

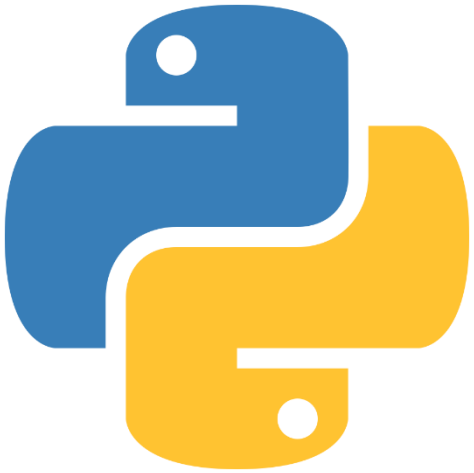
輸入 pip install pillow wordcloud matplotlib jieba numpy 指令安裝套件。

程式碼:

```
#程式
%matplotlib inline
from PIL import Image
import matplotlib.pyplot as plt
from wordcloud import WordCloud
import jieba
import numpy as np
from collections import Counter
# 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)

# 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(一般文字雲)
#my_wordcloud =
WordCloud(background_color='white',font_path=font).generate(words)

#文字雲造型圖片
mask = np.array(Image.open('Python-Symbol.png')) #文字雲形狀

#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(w
ords)

#產生圖片
plt.figure(figsize=(6,6))
```

```
plt.imshow(my_wordcloud)
plt.axis("off")
plt.show()
#存檔
my_wordcloud.to_file('word_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、 在 SQLServer 軟體使用 R 語言繪製文字雲，執行後會自動另存為圖片。(要先設定「啟動外部程式碼」值，在整批執行。)

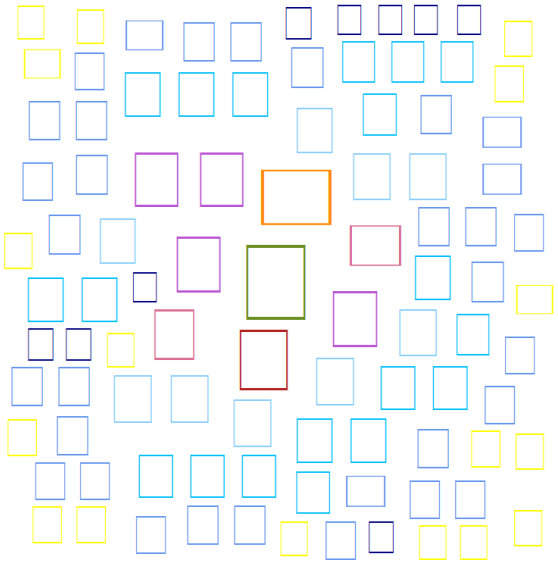
用 2019 版本執行套件時只能在 https://cran-archive.r-project.org/bin/windows/contrib/3.5/網站下載所需要的套面，放在 SQLServer 的 R_SERVER 資料夾上面。

程式碼：

```
EXEC sp_execute_external_script
  @language = N'R',
  @script = N'
library(tm)
library(jiebaR)
library(wordcloud)
library(RColorBrewer)

par(family = "SourceHanSansTW-Regular.ttf")

# 匯入新聞文本
file_path <- "C://SQL0//house_1.txt"
chinese_text <- readLines(file_path, encoding = "UTF-8")

# 斷詞處理
cutter <- worker(user="C://SQL0//dict.txt.big.txt", stop_word =
"C://SQL0//dict.txt.big.txt")
chinese_text <- gsub("[0-9a-zA-Z]+?", "", chinese_text) # 刪除數字和字母
chinese_text <- cutter[chinese_text] # 斷詞

# 計算字頻
freq_ch <- sort(table(chinese_text), decreasing = TRUE)
freq_ch <- as.data.frame(freq_ch)
colnames(freq_ch) <- c("Words", "Freq") # 字頻表

# 設定自訂顏色
customed_colors <- c("#000080", "#ffff00", "#6495ed", "#00bfff",
"#87cefa", "#db7093", "#ba55d3", "#b22222", "#008080", "#ff8c00",
"#6b8e23")
# 生成第一代文字雲
#wordcloud(freq_ch$Words, freq_ch$Freq, min.freq = 2, random.order =
```

```
FALSE, ordered.colors = FALSE, colors = customed_colors)

# 保存文字雲到文件
png("C://SQL0//wordcloud_output.png", width = 800, height = 800)
wordcloud(freq_ch$Words, freq_ch$Freq, scale = c(13, 5), min.freq = 2,
random.order = FALSE, ordered.colors = FALSE, colors = customed_colors)
dev.off()
'
```

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、 在 SQLServer 軟體使用 Python 語言繪製文字雲，執行後會自動另存為圖片。
（要先設定「啟動外部程式碼」值，在整批執行。）

用 2019 版本時執行 pip install pillow==9.4.0 執行安裝套件。需要再 SQLSERVER 的 PYTHOM_SERVER 上執行。

程式碼：

```
EXEC sp_execute_external_script
    @language = N'Python'
    , @script = N'
from PIL import Image
import matplotlib.pyplot as plt
from wordcloud import WordCloud
import pandas as pd
import jieba
from collections import Counter
```

```
# 如果檔案內有一些編碼錯誤，使用 errors="ignore" 來忽略錯誤
with open("C:\\SQL0\\house_1.txt", encoding="Utf-8", errors="ignore")
as f: text = f.read()

# 設定使用 big5 斷詞
jieba.set_dictionary("C:\\SQL0\\dict.txt.big.txt")
wordlist = jieba.cut(text, cut_all=False)
words = " ".join(wordlist)

# 從 Windows 讀取中文字型
font_path = "C:\Windows\Fonts\kaiu.ttf"

#背景顏色預設黑色，改為白色、使用指定圖形、使用指定字體(一般文字雲)
wc = WordCloud(font_path=font_path, width=800, height=800,
background_color="white").generate(words)

#產生圖片
wordcloud = wc.generate(words)
wordcloud.to_file("C:\\SQL0\\wordcloud0.png")

#檢查確定是否有執行成功
print("Word cloud image saved successfully!")
'
```

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
通過這些步驟，我們可以使用 Python 和 R 從 SQL Server 中提取數據，然後生成文字雲並進行情感分析。這些工具和方法能夠幫助我們更好地理解和可視化大量文本數據中的信息。文字雲是可視化的詞頻統計表，常用於具體化和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。