資料探勘導論期末報告

資訊四乙 10927207 蒲品憶

資訊四乙 10927234 吳凱鈺

<https://youtu.be/5EIOLHjttFo>

**題目: Dcard中原版發燒話題類型**

預期成果

* 預期目標

1. 資料集：

寫爬蟲程式，爬Dcard中原版9～11月熱門與最新文章

2.任務：

預處理文章，獲取推文

對文章實行關聯規則分析，找出關鍵字以分析文章類別

統計類別數量，得知中原學生發文喜好

3.評估：

資料視覺化

調整時段長度，觀察不同學期中原學生喜好是否有差異

* 預期用途

希望給學校職員使用，目的是透過Dcard中原版了解學生最近夯什麼，藉由調查可以用於職員在舉辦校內活動時加入最近熱門的主題與元素，提升學生參與活動的意願。

實際結果

* 困難點

一開始預設是利用爬蟲程式蒐集Dcard資料，再將資料轉為DataFrame，最終彙整成Excel表單，成為我們的資料集。但是，在實際執行時，我們發現Dcard其實有很多阻擋爬蟲的機制，在嘗試眾多解決方法後，我們仍無法順利抓取所需的資料，總是會被Dcard發現後阻擋，由於抓取資料集不順利，讓我們的進度遲遲無法向前邁進，又因為有專題繳交的期限壓力，再三考慮下最終我們不得不向老師尋求幫助。

* 困難點解決方案

以下是老師提出的兩種建議:

1. 原始爬蟲不可行的原因，應蒐集、彙整最新可信的文獻或官方文件作為佐證，詳述前因後果，也算是成果之一。
2. 替代方案，盡可能找相近的資料來源作為取代。

* 我們的做法

我們最終決定一人寫一種解決方案

蒲品憶: 原始爬蟲不可行的原因資料蒐集 --------🡪**(方案一)**

吳凱鈺: 以ptt為資料集進行原定相關計畫。 ----🡪**(方案二)**

* **(方案一)爬蟲不可行原因資料蒐集**
  + 一開始我們使用較以前的套件

1.套件**requests**

2.**Dcard API** (2022年都有人在使用)

2.0版本(升級版): <https://www.dcard.tw/service/api/v2/posts>

1.0版本(升級版): <https://www.dcard.tw/_api/posts/>

* 由於Dcard API無法正常載入Dcard因此，之後要進行的requests也無用武之地。

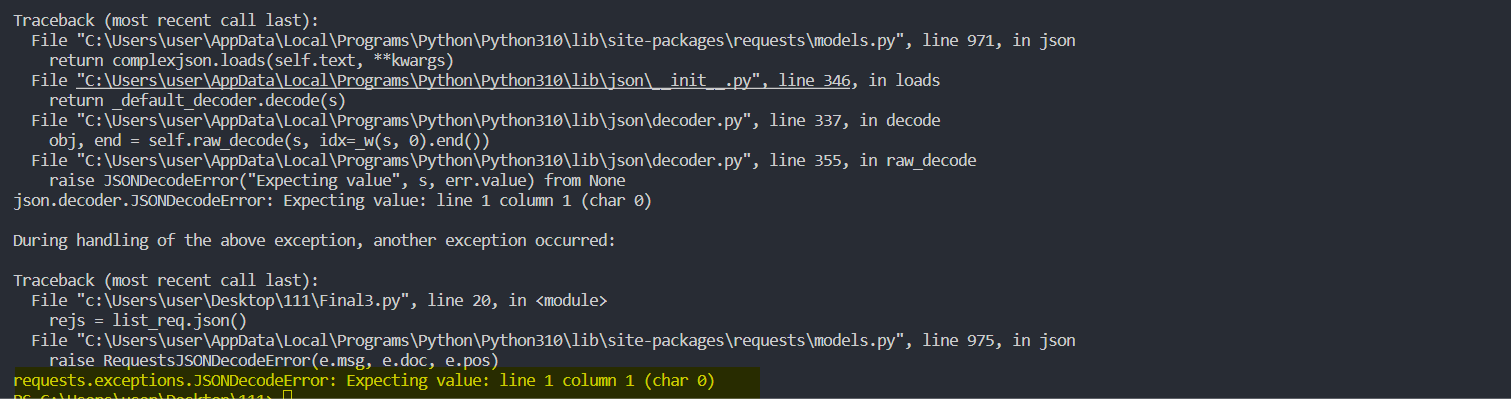
如何發現此錯誤:

LINE1:link = 'https://www.dcard.tw/\_api/posts/' + str(ID)

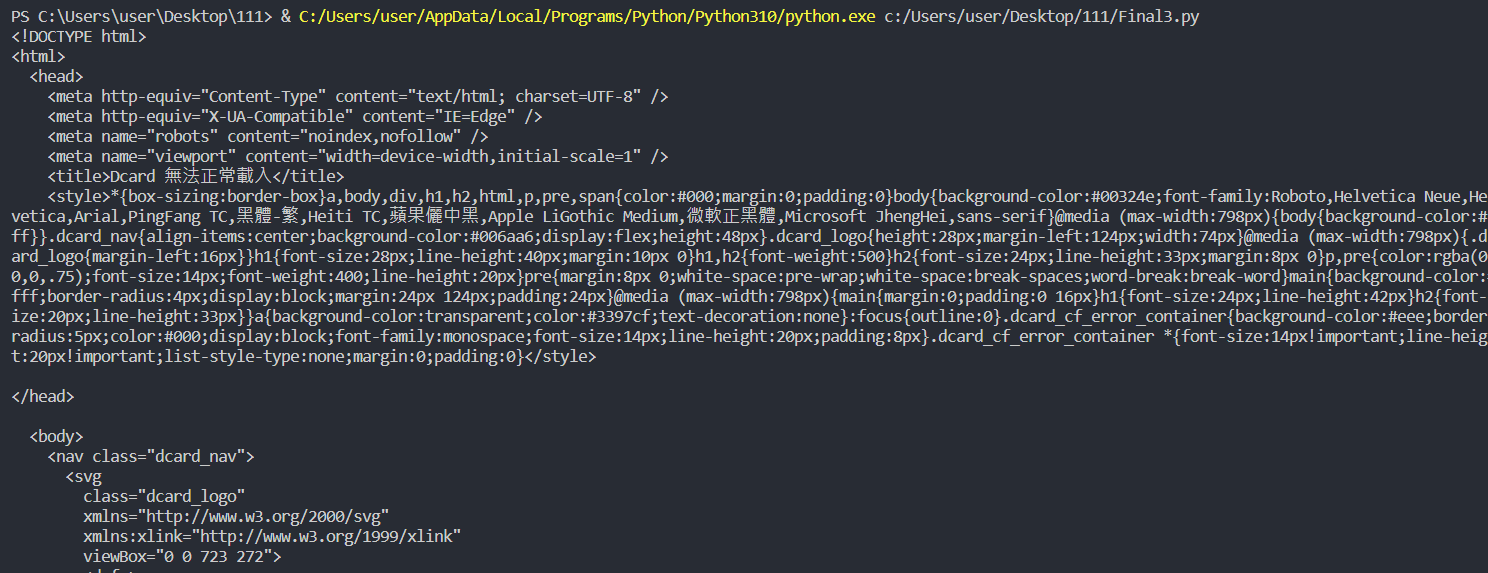
LINE2:requ = requests.get(link)

LINE3:rejs = requ.json()

因為在LINE3轉json會跳出格式錯誤的問題，因此我就去查資料，看看能不能印出再轉成json前requ的內容是什麼(指令:requ.text)，印出來後發現，因為網頁載入錯誤，造成requ.text的內容有些可能不是HTML的格式，導致轉json會出錯，有這項發現我才知道，我沒有成功進入Dcard。



(圖一)轉json的錯誤訊息



(圖二) requ.text內容[html]



(圖三) 網頁版顯示無法正常載入畫面

* + 再來我們使用較新的套件

1. 套件**selenium**這是最近經常被使用來爬蟲的套件

原理: 依我的理解此套件的功能就是在**模擬**使用者在正常情況下上網的一切動作，像是開啟瀏覽器、點網頁、用鍵盤輸入……。

* 問題一: Dcard有看一定數量的文章後，要求登入後才能繼續瀏覽的機制

我在一開始爬少量資料的時候大約8筆資料以內都還可以跑，第九筆以後就需要登入認證，網頁就會卡住無法再往下進行後續操作。



(圖四) 網頁會卡某一頁無法再繼續往下執行

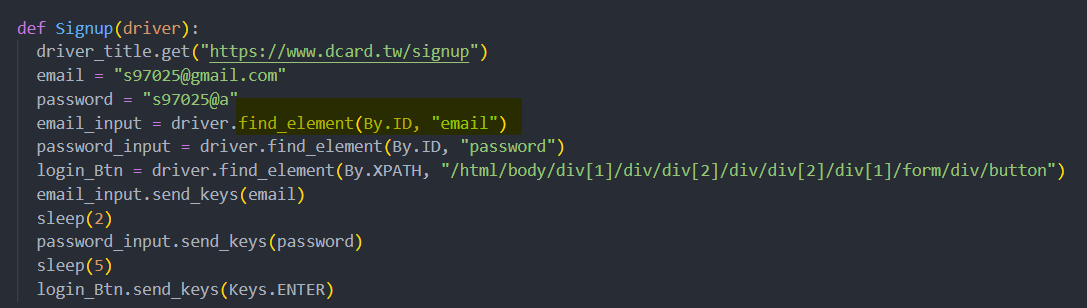
解法: 所以我又去找，如何讓程式自動登入後再開始查詢資料。

一樣使用selenium套件的功能，其中的函式

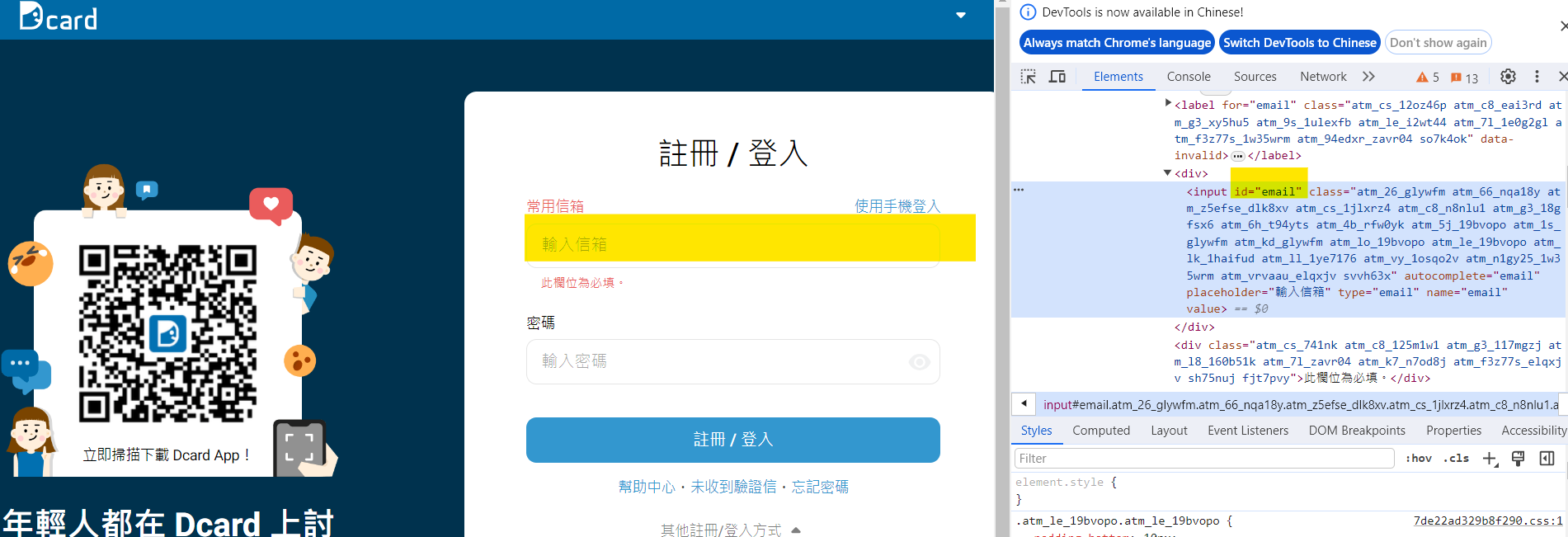
1. Find.element() 找到填入密碼、帳號的框框以及登入按鈕在網頁的位置🡪(圖五)、(圖六) 搭配觀看，塗色部分互相對應

內部填入By.ID、TAG…..[標籤]，以及[標籤]要找的值

1. Send.keys() 相當於使用者利用鍵盤輸入



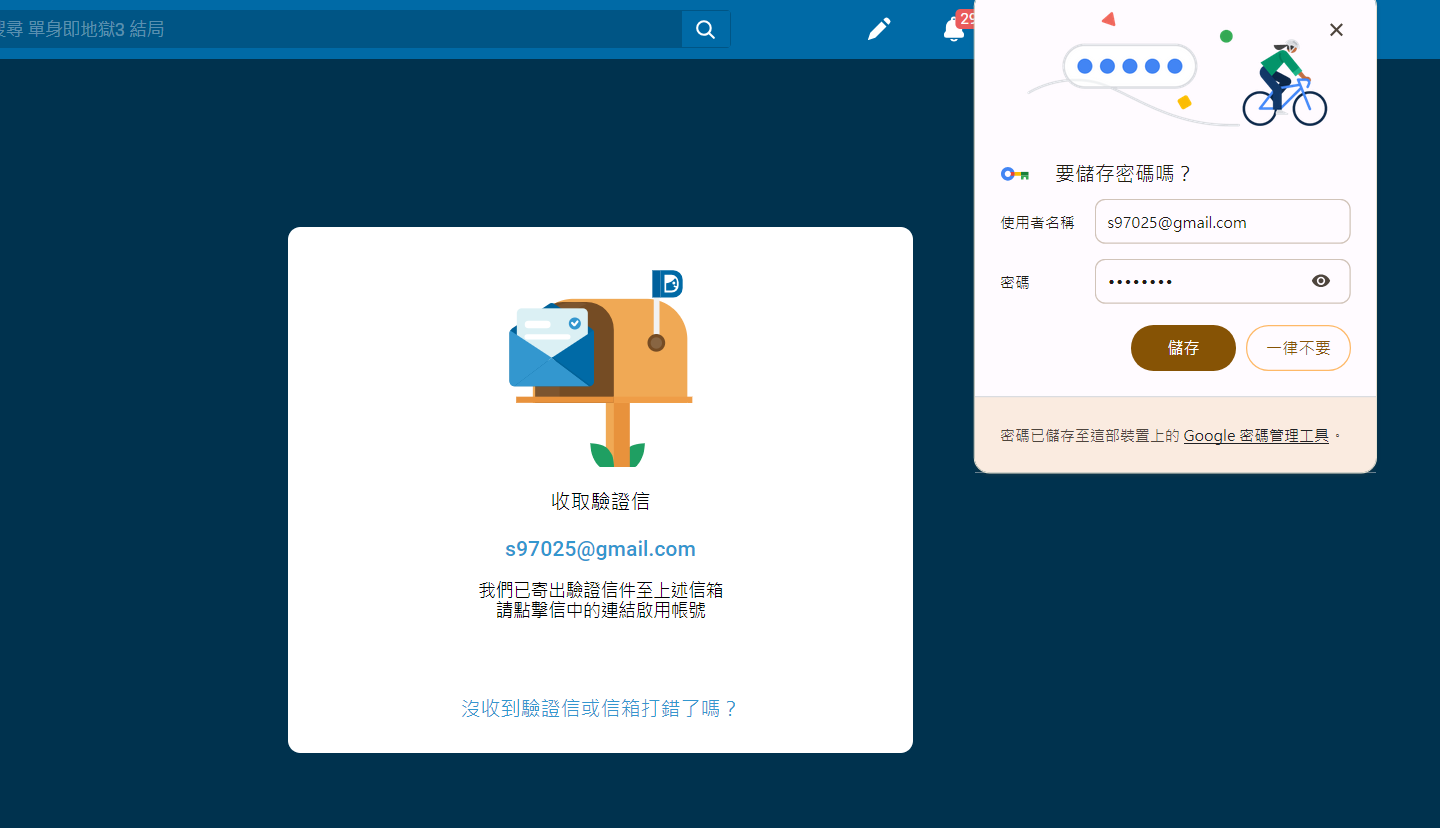
(圖五) 自動登入程式碼



(圖六) 網頁對應位置 [ctrl+shift+I或右鍵>檢查]

* 問題二: 查到如何登入後又有新的問題，可能我之前測試太多次，現在我的Dcard每次登都需要email認證是本人才能登入，到這裡為止算是走到死路了。

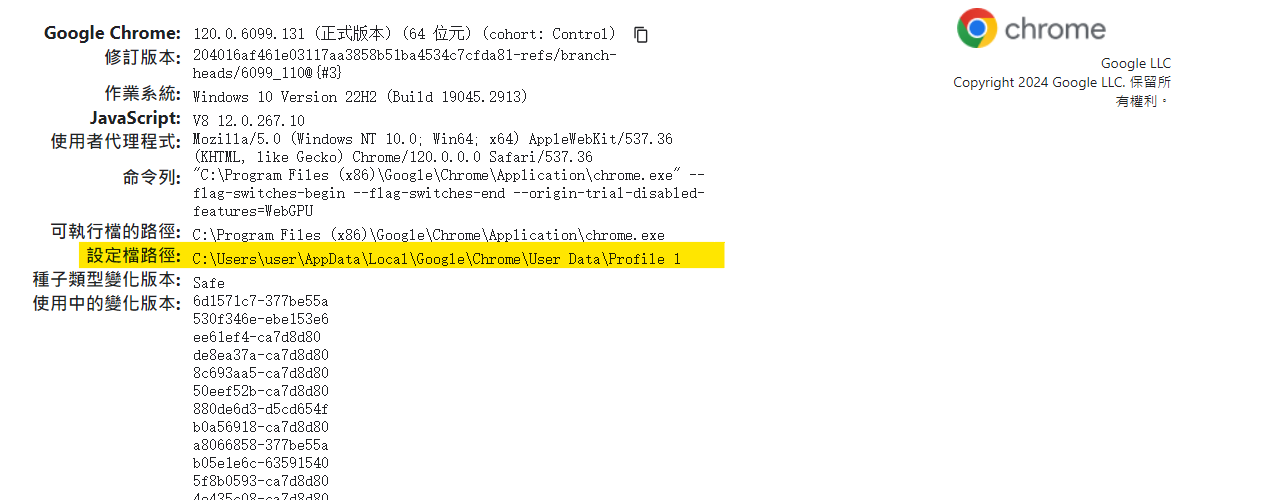
解法: 但是我又想到，如果是用相同的Chrome的使用者，上次登入過後，下一次就不需要再次登入，是否就可以規避掉自動登入會發生的問題。



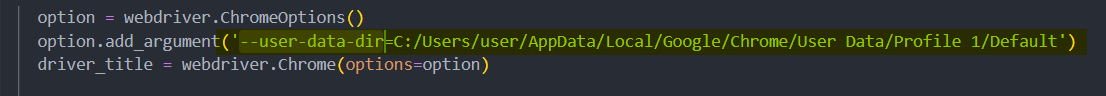
(圖七) 登入認證問題

* 問題三: 相同的Chrome的使用者，是否可以規避掉自動登入會發生的問題。很遺憾，我的實驗結果是否定的，因此最後還是沒辦法把資料從Dcard上面取出來，很可惜。

因為需要知道相同使用者的路徑，因此須利用chrome://version/查看個人路徑，再將路徑貼到--user-data-dir即可確保每次登入，都為相同使用者。



(圖八) 個人路徑



(圖九) 程式碼貼上位置

我觀察到一個蠻有趣的現象，想分享一下。

使用新增的用戶資料資料夾

我們將WebDriver啟動的資料資料夾改為新增的使用者資料路徑，即C:\\Users\\username\\AppData\\Local\\Google\\Chrome\\User Data\\Profile 1，接著啟動就 可以看到開啟了一個新的瀏覽器。

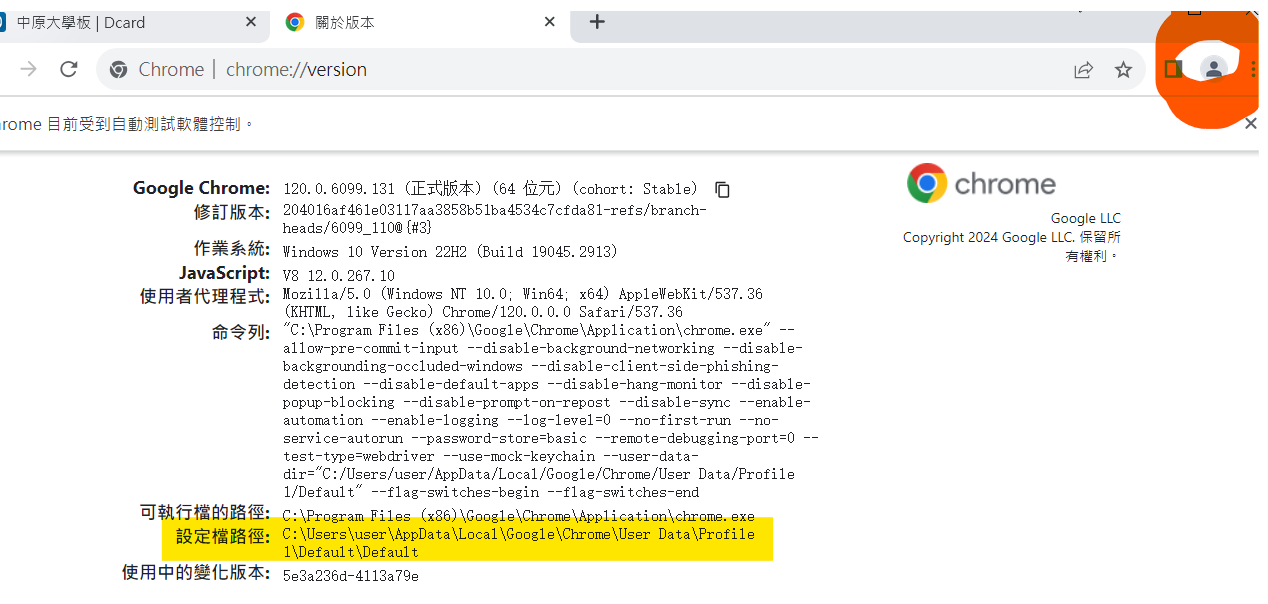
但是你一定注意到了，這裡右上角「個人資料」仍然是空的，而不是剛剛新增用戶的時候設定的頭像。 這是因為Chrome在從命令列第一次啟動的時候，會自動的在當前的使用者資料資料夾下建立\Default資料夾並使用，因此使用的仍然會是一個空的使用者。 所幸這並不影響WebDriver保存用戶資料。

可以在網址列中存取chrome://version查看，個人資料路徑一欄的路徑為C:\\Users\\username\\AppData\\Local\\Google\\Chrome\\User Data\\Profile 1\ \Default。 這時你也會發現瀏覽器在啟動後、執行操作前的停留時間變長了，這是因為瀏覽器是先開啟再載入使用者資料的，因此會比較費時。 也是因為這個原因WebDriver每次都會開啟純淨的瀏覽器來確保運作速度。

白話來說就是: 正常情況下使用【蒲品憶】這位使用者，會顯示我的頭像；但使用driver自動操作則不顯示頭像，但不影響WebDriver保存用戶資料。



(圖十) 正常情形



(圖十) driver自動操作

* Dcard API 失效
* **網路參考網址:** <https://levirve.github.io/blog/2016/Dccard-crawler/>
* **官方回應:**

****

* **(方案二) 以Ptt為資料集進行原定相關計畫**

**題目: PTT各月份最高討論的水果**

預期成果

* 預期目標

1.資料集：

寫爬蟲程式，爬PTT水果版前100頁文章

2.任務：

預處理文章，獲取推文

對文章實行關聯規則分析，找出關鍵字以分析文章類別

統計類別數量，得知PTT各月份最高討論的水果

3.評估：

資料視覺化

* 預期用途

可得知台灣當季暢銷水果類型，以利農夫規劃未來方向。

步驟:

1. 使用爬蟲從PTT上蒐集千則文章
   * 問題一: 有時會出現未滿十八歲禁止進入之彈跳視窗
   * 解決一: 傳送cookie

一張含有 文字, 螢幕擷取畫面, 字型, 行 的圖片

自動產生的描述

圖(十一) cookie設置

1. 從目錄抓取文章標題和其連結
   * 問題二:當文章已被刪除時，在抓取標題時會出現問題
   * 解決二:觀察其位置後發現已刪除之文章的標題與其他普通文章之標題位置不同

一張含有 文字, 字型, 螢幕擷取畫面, 數字 的圖片

自動產生的描述

圖(十二) 判斷正確標題位置

1. 抓取下一頁之連結並繼續重複以上動作100次
2. 切割文章，將其分為作者、看板、標題、日期、內容並儲存
3. 切割文章中的詞語並計算出頻率最高的名詞
4. 從日期中切割出月份並儲存
5. 將名詞、月份、頻率儲存成一個表格
6. 歸納出各月份出現頻率最高的水果並儲存成表格
   * 問題三:圖表的Y軸並非正常 圖(十三)，但同一個code，用其他測試數據卻為正常的 圖(十四)
   * 解決三:目前沒找到解決辦法

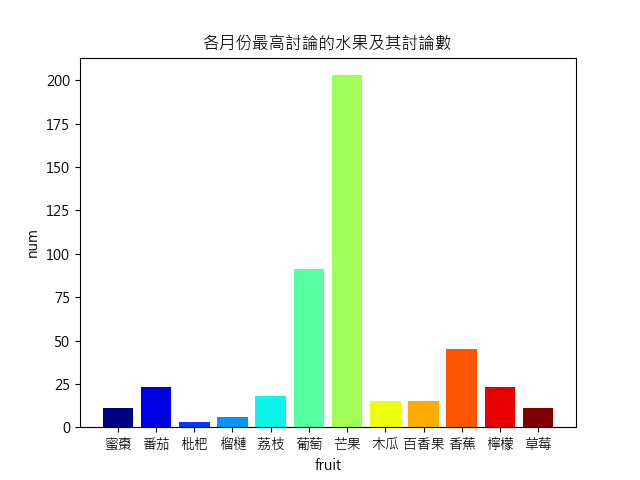
一張含有 文字, 螢幕擷取畫面, 圖表, 繪圖 的圖片

自動產生的描述一張含有 文字, 螢幕擷取畫面, 鮮豔, 圖表 的圖片

自動產生的描述

圖(十三) 錯誤的Y軸 圖(十四)正確的Y軸

* 最終解決畫圖問題:



主要參考資料:

Dcard API: <https://blog.jiatool.com/posts/dcard_api_v2/>

登入帳戶: <https://www.youtube.com/watch?v=tR558fQAHz8>

Selenium 使用 Chrome 瀏覽器 Webdriver:

<https://stackoverflow.max-everyday.com/2018/03/selenium-chrome-webdriver/>

Selenium使用真實用戶Chrome環境: <https://www.cnblogs.com/superhin/p/13954792.html>

<https://developer.aliyun.com/article/1082005>