網際網路資訊檢索-HW1

學號：406410047
系級：資工三
姓名：劉庭聿

I. 開發環境

A. 作業系統

1. Mac OS

B. 使用語言

1. 爬蟲：Python 3.7.4 (BeautifulSoup4)
2. 資料庫：ElasticSearch 6.8.6 + Kibana(可視化)
3. 前端：HTML + CSS (Bootstrap) + Jinja2
4. 後端：Python Flask

II. 爬蟲

A. 爬取 PTT 全站，分為兩個部分。

1. 第一部分：將 PTT 每頁看板的頁面與文章連結爬下來，並且將連結以 md5 進行編碼後當作資料id，將頁面存入board 資料庫，文章連結則存入pool資料庫，為了要減少頻繁對資料庫發送請求，所以使用Batch 的方式將連結打包儲存。
2. 第二部分：將存進資料庫內資料庫以每1000筆文章進行爬蟲，使用異步控制的方式對網頁訪問提升爬取的速度，在解析網頁時使用多線程，增加解析的效率。解析網頁首先使BeautifulSoup 將網頁格式轉換，並且將span.article-meta-value 提取出來，內部存有文章標題、看板分類、作者以及時間的相關資訊。在提取主文方面，將script 、span tag 剔除，script 是將文末的javascript 去除，span 則是把留言去除。取得文章資訊以及內文後，首先判斷文章資訊是否完整，如果文章資訊不完整則將其剔除確保存入資料庫內的資料正確，最後將文章連結以 md5 進行編碼當作id。

III. 資料庫

A. /board/url：儲存看板每個頁面的資料，將url做md5當作id，每次檢查目前爬取的看板頁面是否已經拜訪過，如果以拜訪過就跳過。
B. /pool/url：儲存文章連結，將沒看過的連結存入使用url做md5當作id，確保連結只有一個，在爬取文章的程式會從這個資料庫取得文章連結。
C. /article/art：儲存文章內容，存取文章標題、發文時間、作者、看板分類以及主文。

IV. 前後端

A. 後端部分使用Python Flask套件，將前端所送出的request包裝好後，對文章內文進行搜尋，使用elasticsearch search api，在資料庫進行尋找前50筆搜尋結果。

B. 前端使用Jinja2 將後端response的搜尋結果印出，在使用超連結的方式連到文章頁面。

成果

V. 心得

花了很大部分的時間在處理爬蟲的部分，最一開始的爬蟲版本使用單線程將每個頁面的文章一個一個爬下來，速度很慢。所以添加了異步控制減少訪問網站的等待時間，多線程則是在升解析網頁的速度，比起最初版本的爬蟲時間減少了數倍之多。