Homework5: Query Modeling

> B10632026 吳苡瑄

- 繳交程式
  - Rocchio
  - KL Divergent
- Rocchio
  - 處理讀檔，把所有資料先進來存成我好用的格式 (dict 之類的)
  - 先做一次 VSM
  - 把 VSM 前幾名的 doc 內容當作是新的 query 再做一次 VSM
  - 最後用 Rocchio 的公式得到分數
  - 遇到的困難
    - 因為 vector 會太大，所以我每次都會整理出一個當前 document 和 query 非 0 的 word list，再去做 VSM 運算，這樣其實超級費時間，後來有改成用矩陣和用一些 lib 來做 tfidf，效率有好一點，不過最高分的 (0.53) 還是用舊的 code 跑完的，所以交舊的。
- KL Divergent
  - 一樣先 VSM 取前幾名當新 query
  - 把 doc 和 query 的 tf 做成矩陣，這裡用到 CountVectorizer() 又快又好用
  - 用 SMM 生成 KL 的 P_RM(w|q)
  - 用 KL 距離生成分數
  - 因為 Rocchio 最高只有拿到 0.53 左右，public 看起來連一半的分數都沒有，學長就建議我嘗試 KL Divergent 而且建議我用矩陣做運算，矩陣運算有些地方會繞一點需要用紙筆想一下，不過大致上可以照公式去寫，還有隨時檢查矩陣大小有沒有 match 就還可以
  - 遇到的困難
    - 其實只有在 SMM 的地方會需要用到新的 query tf，其他時候都還是用舊的，這裡卡了很久，因為沒有從講義中理解到這點，出來的一直都試 0.47 這樣的分數，直到問了學長才發現問題，浪費了滿多時間的
    - 直接用 TfidfVectorizer 生成的 VSM 得出的結果也會好一點，應該是 sklearn 內建的公式比較強一點，所以原本直接沿用 hw1 的 VSM 後來經過實驗改掉成新的 TfidfVectorizer 的算法，新舊版本也都一併附上了
- 心得
  只用 Rocchio 其實不難，就算寫再爛也能跑久一點出結果，但分數想要上去就不能只用 Rocchio 有點坑，這次還是要感謝 Hank 學長和電機的游學長，點了我滿多的，然後學到一堆 np 的 sum、reshape 之類的用法瘋掉。
- 最佳參數

```
# Rocchio 0.53
alpha = 1
beta = 0.9
gamma = 0.1
relevant、nonrelevant = 5

# KL 0.58
smm = 0.9
alpha = 0.3
beta = 0.8
gamma = 0.7
top = 5
```