Frequent Pattern Mining

一、應用目標

透過大量的文本資料，能提取出一篇文之中常常一起出現的詞組，透過這些詞組，能用以提取文本關鍵字，並達到分類與減少雜訊的效果。

二、環境要求: Python 3.9 (pkl檔要求，若使用其他格式則3.x即可)

三、事前準備流程:

* 文本爬蟲
  + 重大訊息
* 文字清理
* 斷詞
  + ckip tagger

四、事前準備的檔案:

* 斷詞結果
  + FPM\_input81000.pkl
  + <https://drive.google.com/drive/folders/1nGFJEjWlKoR4dZq2NtsNpk3V97vuX6-u?usp=sharing>

五、介紹:

經過事情準備後，會得到文本的斷詞結果，我在這裡是將結果存成pkl檔。

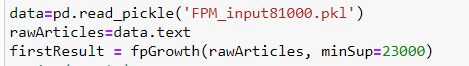
使用FP Growth演算法進行Frequent Pattern Mining。

六、實作流程:

1.資料讀取、FP Growth演算法

將資料讀取並使用FP Growth演算法得到frequent pattern 結果。

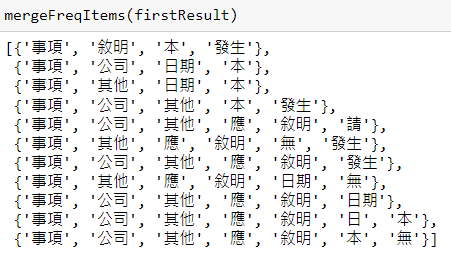
\*minSup為支持度的下限，若詞彙的頻率不及minSup則會被排除，不會被加入FP tree



在這個階段會發現minSup一條界線，大約是資料的1/3~1/4，高於它得到的結果很少或是無。低於它需要耗費相當大的計算時間，且結果不佳。

2.整理結果

整理一下結果，將minSup設為23000得到的結果都是重訊裡非常通用的詞彙組合。



3.去除過於常見的frequent pattern set

因此嘗試將文本中出現這些frequent pattern set的部分去除。

去除我試過兩種方式，一種是用簡化後的結果過濾資料，一種是直接拿FP Growth的結果過濾。



4.再次使用FP Growth

將過濾掉firstResult的文本，再次輸入FP Growth，這次的minSup設為5000，這個數字需要實際測試調整到適合的。



5. 取得結果

secondResult 或FPMresult就是目前的成果了。

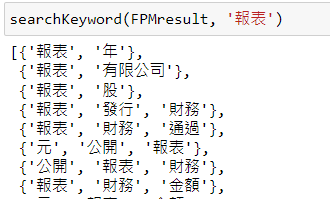


6.成果查看

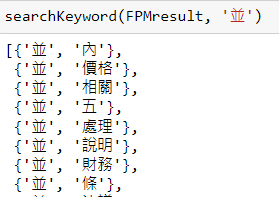
由於結果資料量龐大，我挑選幾個詞彙來察看結果。

可以發現成功的取得Frequent Pattern，但同時也存在不易去除的雜訊。

雜訊的情況還需要修改算法或使用其他的工具或演算法嘗試濾除。



雜訊:



結果查看:

example 1



example 2



七、Source code

[stock/Frequent\_Pattern\_Mining.ipynb at main · samwang1228/stock (github.com)](https://github.com/samwang1228/stock/blob/main/%E6%B2%9B%E9%8C%A1/Frequent_Pattern_Mining.ipynb)