Information Retrieval – HW2

B10615046 柯元豪

**實作環境**

Jupyter notebook (Python 3)

**使用的套件**

numpy, pandas, math, nltk

**資料前處理**

指定document, query的目錄，用nltk的PlaintextCorpusReader去目錄下存取所有檔案，之後將檔案分別存至對應的list中

**參數調整**

1. TF 只計算詞的出現次數
2. IDF 做 smooth -> log(N – df + 0.5 / df + 0.5)
3. BM25參數設定： K1 = 3, K3 = 1000, b = 0.85 (實驗後最佳參數組合)
4. 實作BM25L，在計算document weight時，加上δ，δ值設為0.5

參考論文：<http://www.cs.otago.ac.nz/homepages/andrew/papers/2014-2.pdf>

**運作原理**

1. 依序讀取query
2. 取得此query 中所有詞與文件的BM25分數，存成對應矩陣
3. 將此query所有詞的分數加總後排序
4. 將document排序並匯出

**心得**

這次實作比起上次vsm相對輕鬆一點，因為程式架構上只有最後vsm做相似度比較的部分不同，改成取得分數後直接做ranking，較煩瑣的部分應該還是調整參數，除了找大眾一點的參數組合實驗之外，還有參考幾篇較短的論文暸解一下各種參數對於演算法的影響，最後的確也發現了bm25L的效果來得好很多，實驗過程也是很有收穫的。