IR HW2 資管碩二 R05725034 張鑑霖

1. 執行環境 & 作業系統

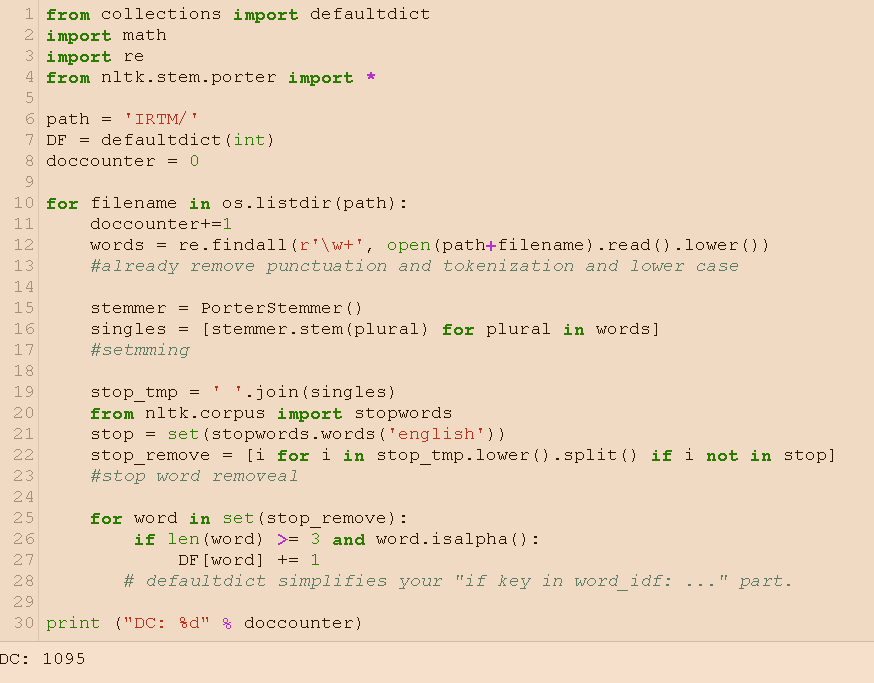
Jupyter & win7

1. 程式語言

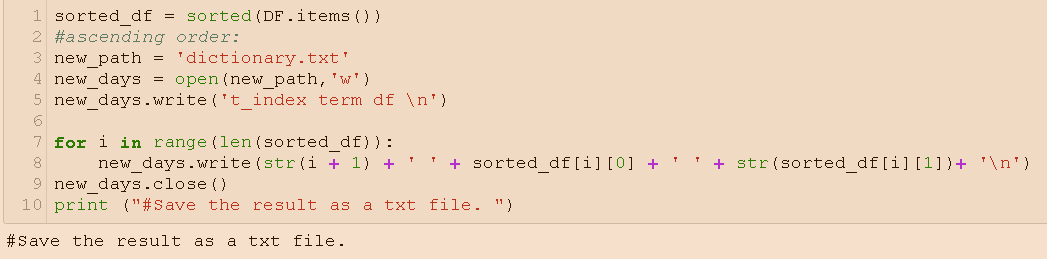
Python3

1. 作業處理邏輯說明
   1. Construct a dictionary:
      1. 讀入每個在資料夾IRTM的檔案

並去除標點、tokenization、小寫化、stemming、去除stop word，並記錄有幾筆文件(1095筆)，如果字長度大於3和是英文，就把此文字的計數器+1，因為用的是set，所以處理時每份文件不會計算重複的字。

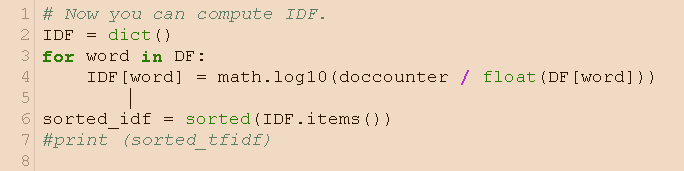


* + 1. 輸出

把紀錄的文字 - 計數依照文字排序，並存入dictionary.txt  


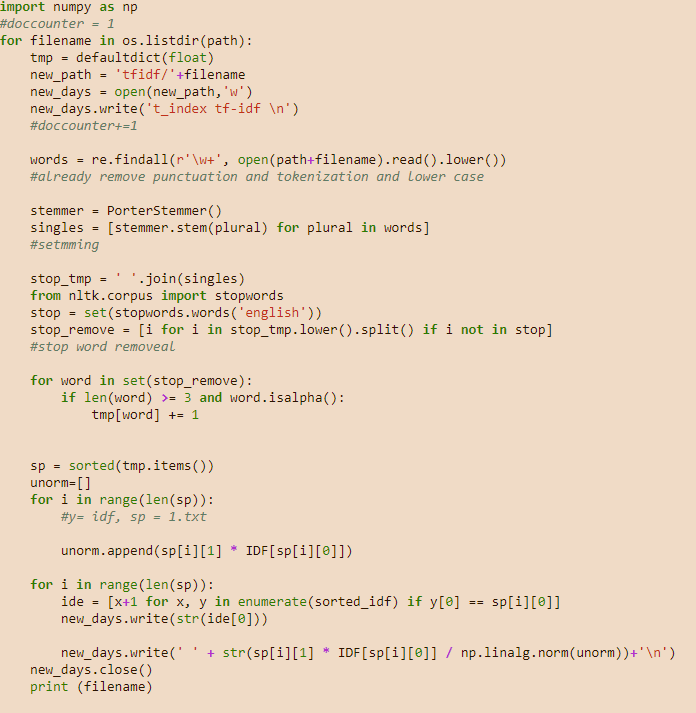


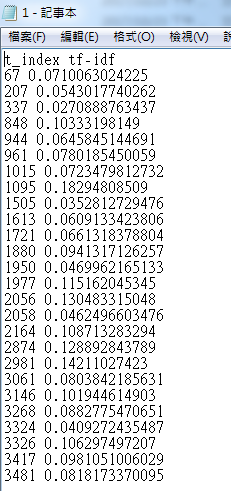
* 1. Tfidf unit vector
     1. 計算每個文字的idf(log以10為底)，並排序



* + 1. 開起每個在資料夾IRTM的檔案，做一樣的前處理

得到每個TF，每個TF再乘上剛剛相對應的IDF，再標準化即可得到結果，檔案存於tfidf資料夾內。





* 1. function *cosine*
     1. 輸入兩個文件名計算cosine similarity，先讀入問題”乙”的檔案，並計算兩個檔案的norm當分母，而分子只需要算兩份文件都有出現的term，並把他們的tfidf相乘後相加即可，例如cosine('1.txt', '2.txt') =0.149421757697

1. 任何在此作業中的心得

有些地方會執行緩慢，可能有地方可以優化執行效率，是我可以努力的目標，而且在處理這三小題中，有些地方感覺用到的變數也是有重複，如果能更佳善用可以節省記憶體空間，也可以運算得更快速