HW4-Sentiment Analysis

* 本次作業將針對Twitter資料進行情緒分析(Sentiment Analysis)，預測一則推特之情緒為正面或負面，資料集共有兩個:

1. "training\_label.txt"為根據表情符號(emoticon)進行自動標籤的訓練資料，共有20萬筆。
2. "testing\_label.txt"為人工標籤資料，共90筆。

檔案中的"+++$+++"和"#####"為Label(1或0)與推特內容的分割符號，1代表正面情緒，0代表負面情緒。請分別利用RNN與LSTM對"training\_label.txt"的資料建模，對"testing\_label.txt"測試計算Accuracy，可使用Keras(建議)或Tensorflow來完成本次作業。

* 作業流程：

1. 資料前處理

a. 讀取"training\_label.txt"與"testing\_label.txt"並利用分割符號切割字串、建立train&test之DataFrame

b. 建立Token：利用Tokenizer建立字典並將文字轉換為數字

c. Padding：將各筆資料之長度固定(截長補短)

2. 建模

a. 分別用RNN與LSTM對training\_label.txt的資料進行建模，可自行設計神經網路的架構(epoch設10即可)

b. 加入Dropout Layer設定Dropout參數(建議0.7)進行比較

c. plot出訓練過程中的Accuracy與Loss值變化 \*hint: 抓出history回傳的'acc','val\_acc','loss','val\_loss'四項數值

3. 評估模型

a. 利用"testing\_label.txt"的資料對2.所建立的模型進行測試，並計算Accuracy

* 請繳交程式碼與討論報告，討論報告須包含:

1. RNN與LSTM在有無Dropout的設計下(e.g. 無Dropout & Dropout=0.7)的訓練Accuracy、Loss圖

以及3.評估模型的Accuracy值

2. 發現與討論: 例如RNN與LSTM的準確度比較、加入Dropout後的Loss圖變化、是否有Overfitting/Underfitting的情況發生等

Loss & Accuracy圖範例:

https://imgur.com/oVfdwa0

P.S.

1. Keras安裝流程可參考工具書第4章(CPU版本即可)

2. 作業流程可參考"CNN\_RNN\_LSTM實作.pptx"的範例或工具書第14章

3. 參考論文:Twitter Sentiment Classification using Distant Supervision