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AdaBoost與xgboost對"training_label.txt"的資料建模，並用"testing_label.txt"進行測試

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

註：因資料量龐大，後續使用 tf-idf 轉向量後會有記憶體不足的問題，同學可以挑選適量data (ex. 前一萬筆) 進行建模，並在文件中說明即可

b. 去除停頓詞stop words

可參考：

- sklearn.feature_extraction.text.CountVectorizer
  https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html
- 自訂stop words
  https://stackoverflow.com/questions/52712254/how-to-eliminate-stop-words-only-using-scikit-learn

c. 文字探勘前處理，將文字轉換成向量，像是常見的方法 tf-idf、word2vec...等

可參考：

- sklearn.feature_extraction.text.TfidfVectorizer
  https://scikit-learn.org/stable/modules/classes.html#module-sklearn.feature_extraction.text
- Word2vec
  https://radimrehurek.com/gensim/models/word2vec.html

2. 建模：分別使用以下兩種模型

- AdaBoost
  https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.AdaBoostClassifier.html
- xgboost
  https://xgboost.readthedocs.io/en/latest/python/python_intro.html#install-xgboost

3. 評估模型

利用"testing_label.txt"的資料對2.所建立的模型進行測試，並計算Accuracy、Precision、Recall、F-measure