[資料前處理]

```
train = pd.DataFrame(columns = ['mood','sentence'])
trainfile = open('training_label.txt','r')
line = trainfile.readline()
i = 0
while line:
    if line != '\n':
        i+=1
        temp = line.split("+++$+++", 1)
        new = pd.DataFrame({'mood':temp[0],'sentence':temp[1]},index=[1])
        train = train.append(new,ignore_index=True)
    if i == 100000 : break
    line = trainfile.readline()
trainfile.close()
```

- 切割字串，存成dataframe

以#####或+++$+++切割字串，前面儲存到mood column，後面儲存到sentence column。
test取全部90筆資料，train則嘗試取100000筆資料。

- 停用字

有嘗試直接使用nltk的停用字，但透過觀察停用字發現他會刪掉像no ,not ,don't等可能對於負面情緒有意義的字眼。因此我透過countvectorizer觀察vocabulary的詞頻來刪減重複次數多且較無意義的字，以下是我刪除的停用字：

```
stop_words = [',','.','..','...',"'",'`',':','1','2','3','4','5','6','7','8','9','0',
              '00','000','0000','000pv','day','so','all','up','got','today','from','one',
              'a','b','c','d','e','f','g','h','i','j','k','l','m','n','o','p','q','r','s',
              't','u','v','w','x','y','z','im','you','your','u','ur','they','an','him',
              'we','he','she','me','my','his','her','it','is','am','are','was','were',
              'btw','thing','be','have','has','had','do','does','did','ll','re','ve',
              'this','that','there','say','says','said','guys','to','for','in','at',
              'when','the','on','its','and','in','of','some','someone','before','after',
              'with','been','being','which','them','their','our','us','left','10','these',
              '30','site','online','12','da','room','20','sometimes','11','sat','15','google'
              '24','info','sit','web','website','09','17','18','33','50','session','16','21',
              '25','b4','pm','13','333','45','70','2009','month','yr','yrs','23',
              'txt','06','22','26','29','31','37','80','07','19','28',
              '32','34','35','36','38','48','52','69','79','89','95','97']
```

- 文字轉向量

```
tv = TfidfVectorizer(use_idf=True, smooth_idf=True, norm=None)
train_fit = tv.fit_transform(train_x)
test_fit = tv.transform(test_x)
```

使用TfidfVectorizer，以train_x也就是刪掉停用字後的sentence資料去fit跟transform，test_x再透過train_x fit好的model轉換，才能保證train_fit跟test_fit的feature一樣，之後建模才不會有問題。

[模型結果比較]

- 未使用任何停用字處理

```
AdaBoost
              precision    recall  f1-score   support

           0       0.70      0.76      0.73        37
           1       0.82      0.77      0.80        53

    accuracy                           0.77        90
   macro avg       0.76      0.77      0.76        90
weighted avg       0.77      0.77      0.77        90

XGBoost
              precision    recall  f1-score   support

           0       0.64      0.73      0.68        37
           1       0.79      0.72      0.75        53

    accuracy                           0.72        90
   macro avg       0.72      0.72      0.72        90
weighted avg       0.73      0.72      0.72        90
```

- 使用nltk停用字

```
with nltk stop words
AdaBoost
              precision    recall  f1-score   support

           0       0.67      0.59      0.63        37
           1       0.74      0.79      0.76        53

    accuracy                           0.71        90
   macro avg       0.70      0.69      0.70        90
weighted avg       0.71      0.71      0.71        90

XGBoost
              precision    recall  f1-score   support

           0       0.61      0.51      0.56        37
           1       0.69      0.77      0.73        53

    accuracy                           0.67        90
   macro avg       0.65      0.64      0.65        90
weighted avg       0.66      0.67      0.66        90
```

可以看出使用nltk停用字甚至比沒用的效果還差，推測可能的原因就像上面所提到的，nltk的停用字可能包含對於情感分析來說有意義的字。

- 自訂停用字

```
with costom stop words
AdaBoost
              precision    recall  f1-score   support

           0       0.71      0.73      0.72        37
           1       0.81      0.79      0.80        53

    accuracy                           0.77        90
   macro avg       0.76      0.76      0.76        90
weighted avg       0.77      0.77      0.77        90

XGBoost
              precision    recall  f1-score   support

           0       0.67      0.65      0.66        37
           1       0.76      0.77      0.77        53

    accuracy                           0.72        90
   macro avg       0.71      0.71      0.71        90
weighted avg       0.72      0.72      0.72        90
```

結果跟沒使用停用字差不多，甚至還差了一點，推測原因可能是tfidf的idf部分本來就會把出現頻率高的字降低權重，就類似於刪除停用字的效果，但原因也可能是刪除的停用字還不夠多，有些無意義的字可能因為出現少次所以會讓那個字對於某個句子的代表性高，事實上對於情感分析卻是無意義的。

- 自訂停用字＆調整參數

```
with costom stop words and tuned parameter
AdaBoost
              precision    recall  f1-score   support

           0       0.78      0.76      0.77        37
           1       0.83      0.85      0.84        53

    accuracy                           0.81        90
   macro avg       0.81      0.80      0.80        90
weighted avg       0.81      0.81      0.81        90

XGBoost
              precision    recall  f1-score   support

           0       0.79      0.84      0.82        37
           1       0.88      0.85      0.87        53

    accuracy                           0.84        90
   macro avg       0.84      0.84      0.84        90
weighted avg       0.85      0.84      0.84        90
```