情緒分析 (sentiment analysis) 又稱為意見探勘 (opinion mining).

是使用 "自然語言處理", 文字分析等方法, 找出作者某些話題上的態度, 情感, 評價或情緒. 情緒分析的商業價值, 可以提早得知顧客對公司或產品觀感, 以調整銷售策略方向.

[IMDb](http://www.imdb.com/) 網路資料庫 (Internet Movie Database), 是一個電影相關的線上資料庫. IMDb 開始於 1990 年, 自 1998 年起成為亞馬遜旗下的網站, 至今已經累積大量的電影資訊. IMDb 共收錄了四百多萬作品資料.   
  
IMDb 資料集共有 50,000 筆 "影評文字", 分為訓練資料與測試資料各 25,000 筆, 每一筆 "影評文字" 都被標記成 "正面評價" 或 "負面評價". 我們希望能建立一個模型, 經過大量 "影評文字" 訓練後, 此模型能用於預測 "影評文字" 是 "正面評價" 或 "負面評價". 如下面深度學習模型用於辨識 IMDb "影評文字", 可分為訓練與預測階段:

很多線上社群網站會蒐集使用者的資料，並且分析使用者行為，像是知名的Facebook在前幾年開始做「情緒分析(sentiment analysis)」，它是以文字分析、自然語言處理NLP的方法，找出使用者的評價、情緒，進而預測出使用者行為來進行商業決策，像這樣一連串利用情緒分析帶來的商業價值是相當可觀的。

1.使用urllib下載資料集

2.使用tarfile解壓資料集

3.使用re書寫規則運算式，替換文本中的格式符

4.使用Tokenizer去建立字典

5.使用sequence去變換文本長度，短的補0，長的截取

6.使用Embedding層將數位清單轉換為向量清單

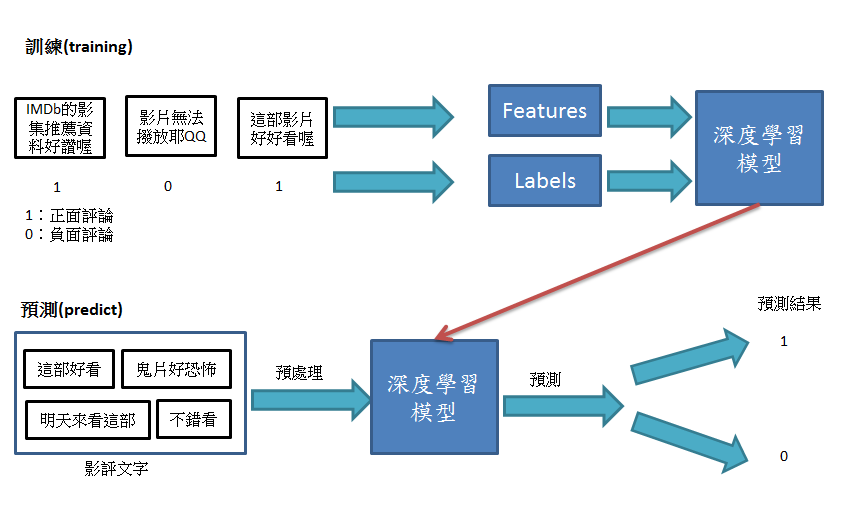
1.keras的simpleRNN函數使用的RNN結構暫時不清楚

2.keras嵌入層從數字清單轉化為向量清單中間過程暫時不清楚

以[IMDb網路電影影評](https://www.imdb.com/)資料集作範例。

建立一個模型，要有大量的訓練(training)後來做預測(predict)，

在訓練時蒐集正反兩面的資料的特徵值及屬性來建立深度學習模型，再利用預測資料將資料做預處理找出特徵值，透過實作完成的深度學習模式做預測，並且計算出預測結果與真實值得差異，計算出建立完成的深度學習模型的準確值如何。



1. 先下載這是範例需要的資料檔[IMDb的資料集](http://ai.stanford.edu/~amaas/data/sentiment" \t "_blank)，並匯入模組，進行檔案下載、確認檔案、解壓縮

<http://ai.stanford.edu/~amaas/data/sentiment/>

1. 讀取IMDb資料

from keras.datasets import imdb

from keras.preprocessing import sequence

from keras.preprocessing.text import Tokenizer

*#將html的標籤刪除*

import re

def rm\_htmltags(text):

re\_tag = re.compile(r'<[^>]+>')

return re\_tag.sub('', text)

*#讀取IMDb的檔案目錄*

import os

def read\_files(filetype):

path = "IMDb/aclImdb/"

file\_list=[]

positive\_path=path + filetype+"/pos/"

for f in os.listdir(positive\_path):

file\_list+=[positive\_path+f]

negative\_path=path + filetype+"/neg/"

for f in os.listdir(negative\_path):

file\_list+=[negative\_path+f]

print('read',filetype, 'files:',len(file\_list))

all\_labels = ([1] \* 12500 + [0] \* 12500)

all\_texts = []

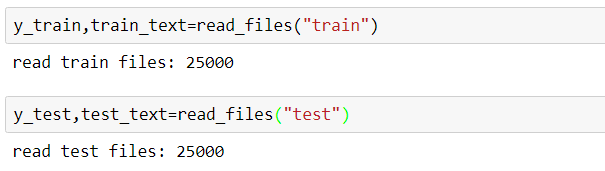
for fi in file\_list:

with open(fi,encoding='utf8') as file\_input:

all\_texts += [rm\_tags(" ".join(file\_input.readlines()))]

return all\_labels,all\_texts

*#查看IMDb的檔案目錄*



1. 使用Tokenizer建立token

*#先讀取所有文章建立dic，限制dic數量為nb\_words=2500*

token = Tokenizer(num\_words=2500)

token.fit\_on\_texts(train\_text)

*#查看文字index的屬性*

