# Homework 5 Report - Text Sentiment Classification

學號：b05902013 系級：資工二 姓名：吳宗翰

## 1. 請說明你實作的 RNN model，其模型架構、訓練過程和準確率為何？ (1%)

**模型架構**

```
### model 1 -> 1 LSTM + 1 Dense ###

Input (30, 200, 1)
LSTM(256, tanh)
Dense(64) ReLU + Dropout(0.5)
Output (7) (Sigmoid)

### model 2 -> 2 LSTM + 1 Dense ###

Input (30, 200, 1)
LSTM(128, tanh, return sequence)
LSTM(128, tanh)
Dense(64) ReLU + Dropout(0.5)
Output (7) (Sigmoid)

### model 3 -> 2 LSTM + 1 Conv + 1 Dense ###

LSTM(128, tanh, return sequence)
LSTM(128, tanh, return sequence)
Conv(filter=8, kernel=(2, 2)) + BN + LeakyReLU
Dense(128) ReLU + Dropout(0.5)
Output (7) (Sigmoid)

### Average Blending ###
average 全部的y再用0.5當成threshold就可以區分0, 1
```

**訓練參數與預處理**

**hyper parameter**

1. Optimizer: Adam, learning rate等等參數都按照keras預設值
2. loss function: Binary Cross Entropy
3. epoch: 每個model各12個epoch
4. batch size: 50
5. 其他參數：BN momentum 0.5，其他均使用keras default

**data preprocessing**

1. 先把句子中重複的兩個字元壓縮成一個(ex: happyyyyyyyy -> hapy)
2. 人工replace一些字串(ex: i ' m -> im)
3. 使用training data以及nolabel的semi supervised data用gensim先word2vec，embed成200維，min_count=5(預設)
4. 取每一句的30個word作為predict，如果遇到不在字典裡面的字就直接忽略他

**準確率**

| Public Score | Private Score |
| --- | --- |
| 0.83188 | 0.82952 |

備註：單用model 1其實就已經過Strong Baseline了，只是不可否認的Blending確實進步了不少

## 2. 請說明你實作的 BOW model，其模型架構、訓練過程和準確率為何？(1%)

**模型架構**

```
Input(200)
Dense(128) + ReLU + Dropout(0.4)
Dense(64) + ReLU + Dropout(0.4)
Output(1) (Sigmoid)
```

**訓練參數與預處理**

**hyper parameter**

1. Optimizer: Adam, learning rate等等參數都按照keras預設值
2. loss function: Binary Cross Entropy
3. epoch: 5個epoch(容易overfit)
4. batch size: 50
5. 其他參數：均使用keras default

**data preprocessing**

同第一題，唯一不同就是把一整句中所有word的gensim後兩百維的東西加起來

**準確率**

| Public Score | Private Score |
| --- | --- |
| 0.78426 | 0.78378 |

## 3. 請比較bag of word與RNN兩種不同model對於"today is a good day, but it is hot"與"today is hot, but it is a good day"這兩句的情緒分數，並討論造成差異的原因。(1%)

**情緒分數比較**

- BOW model: 兩句都是0.78425，而且是相同的結果 (兩句都是正面情緒)
- RNN model: 第一句0.40153，第二句0.89371以上 (第一句低信心表示負面，第二句正面)

**差異探討**

- 在BOW model中，不意外的因為他把所有字加起來，讓兩個句子的Input向量相同，得到相同的預測結果(在這裡都是正向)
- 在RNN model中，語序就顯得重要，而最後也就得到兩個不同的結果，也比較make sense

## 4. 請比較"有無"包含標點符號兩種不同tokenize的方式，並討論兩者對準確率的影響。(1%)

**Tokenize 比較**

| Tokenize Method | Public Score | Private Score |
|---|---|---|
| 包含標點符號 | 0.80240 | 0.80034 |
| 清掉標點符號 | 0.80671 | 0.80369 |

**差異探討**

在這題中為了比較這兩點而使用raw data並且用了個簡單的network進行預測，在這題中可以看到清掉標點符到的模型有高一點點的performance，猜測差異主要來自於標點符號的使用習慣(有些人用驚嘆號，有些人用句號，拿掉的話可以讓variance下降)。不過也可以從這裡看出在這種task中data preprocessing還是重要的。

## 5. 請描述在你的semi-supervised方法是如何標記label，並比較有無semi-surpervised training對準確率的影響。(1%)

**如何標記label**

1. 三個model(第一題有提到)進行投票，當他們三者都預測是同一個label的data就會被我append進去原始資料中繼續訓練
2. 大致還可以挑出一半的資料(超過70w資料)

**準確率**

| Training method | Public Score | Private Score |
|---|---|---|
| Supervised | 0.83188 | 0.82952 |
| Semi-Supervised | 0.82540 | 0.82439 |

**討論原因**

在本題中光是用Supervised Learning就很容易overfit了，再加上Semi-Supervised進來的東西不一定乾淨，導致結果不增反減。研判可能還需要更多的model進行voting或者是還是要設定一個threshold(例如0.8以上)割出來比較好