

1. (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？
(Collaborators:)

總共用了兩個 model 做 ensemble，以下只列出其中一個模型

句子的最大長度為 38，embedding 的維度為 200

模型架構：

| Layer (type) | Output Shape | Param # |
|----------------------------------|-----------------|----------|
| embedding_5 (Embedding) | (None, 38, 200) | 48212400 |
| lstm_9 (LSTM) | (None, 38, 256) | 467968 |
| lstm_10 (LSTM) | (None, 256) | 525312 |
| dense_5 (Dense) | (None, 2) | 514 |
| Total params: 49,206,194 | | |
| Trainable params: 993,794 | | |
| Non-trainable params: 48,212,400 | | |

訓練過程 & 準確率：

Train on 180000 samples, validate on 20000 samples

Epoch 1/2

loss: 0.4682 - acc: 0.7779 - val_loss: 0.4341 - val_acc: 0.8026

Epoch 2/2

loss: 0.3649 - acc: 0.8375 - val_loss: 0.4345 - val_acc: 0.8052

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

模型架構：

| Layer (type) | Output Shape | Param # |
|-----------------------------|--------------|---------|
| dense_1 (Dense) | (None, 128) | 1280128 |
| dropout_1 (Dropout) | (None, 128) | 0 |
| dense_2 (Dense) | (None, 128) | 16512 |
| dropout_2 (Dropout) | (None, 128) | 0 |
| dense_3 (Dense) | (None, 128) | 16512 |
| dropout_3 (Dropout) | (None, 128) | 0 |
| dense_4 (Dense) | (None, 128) | 16512 |
| dense_5 (Dense) | (None, 128) | 16512 |
| dense_6 (Dense) | (None, 1) | 129 |
| Total params: 1,346,305 | | |
| Trainable params: 1,346,305 | | |
| Non-trainable params: 0 | | |

訓練過程 & 準確率：

Train on 180000 samples, validate on 20000 samples

Epoch 1/3

loss: 0.4942 - acc: 0.7601 - val_loss: 0.4527 - val_acc: 0.7892

Epoch 2/3

loss: 0.3986 - acc: 0.8204 - val_loss: 0.4554 - val_acc: 0.7912

Epoch 3/3

loss: 0.3287 - acc: 0.8551 - val_loss: 0.4738 - val_acc: 0.7877

3. (1%) 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。

(Collaborators:)

bag of word 的值越接近 1 為正面，越接近 0 為負面

RNN 目標值我用 one-hot-encoding 表示，因此比較 label 0 與 label 1 的分數看何者較大就是哪種情緒

| | today is a good day, but it is hot | today is hot, but it is a good day |
|-------------|------------------------------------|------------------------------------|
| bag of word | 0.80802143 | 0.80802143 |
| RNN label 0 | 0.882416 | 0.038436 |
| RNN label 1 | 0.117584 | 0.961564 |

由於 bag of word 沒有順序的概念，因此兩句訓練出來的分數一樣，而 RNN 模型有考慮順序，因此根據轉折語氣來判斷，today is a good day, but it is hot 是負面，today is hot, but it is a good day 是正面的判斷的確比較好

4. (1%) 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。

(Collaborators:)

利用 tokenize 的 filters 參數來過濾標點符號，無包含標點符號時，RNN 模型的 val_acc 為 0.8115，而有包含標點符號時，RNN 模型的 val_acc 為 0.8187，因此在我的模型中，將標點符號包含進來會得到較好的準確率，這可能跟有些標點符號可以表達情緒（問號與驚嘆號），以及句子結構有關

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

(Collaborators:)

測試 semi-supervised 時只用單一模型做測試，先訓練好一個 val_acc 為 0.8091 的 RNN 模型，利用此模型對 training_nolabel.txt 做預測，並且設定一個門檻值，只要 RNN label 0 或 RNN label 1 的分數有高於 0.85 就將此筆資料貼上對應的標籤加入訓練資料中，最後用新的訓練資料重新訓練，最後得到的 val_acc 為 0.8052，因此我的模型使用此種方法並沒有增加效能