

ML HW5 Report

學號：B05902052 系級：資工二 姓名：劉家維

以下的準確率若未特別說明，皆是以 public/private 的形式表示。

以下的情緒分數，皆為越接近 1 則表示越正面，越接近 0 則越負面。

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？

答： 架構：

Layer (type)	Output Shape	Param #
bidirectional_1 (Bidirection	(None, 37, 256)	439296
bidirectional_2 (Bidirection	(None, 128)	164352
batch_normalization_1 (Batch	(None, 128)	512
dense_1 (Dense)	(None, 256)	33024
dropout_1 (Dropout)	(None, 256)	0
dense_2 (Dense)	(None, 128)	32896
dropout_2 (Dropout)	(None, 128)	0
dense_3 (Dense)	(None, 64)	8256
dropout_3 (Dropout)	(None, 64)	0
dense_4 (Dense)	(None, 2)	130
Total params: 678,466		
Trainable params: 678,210		
Non-trainable params: 256		
None		

(summary 前兩層是 Bidirectional 的 LSTM)

loss function: categorical_crossentropy, optimizer: adam

epoch: 20 (但只存 validation accuracy 最高的)

準確率：0.82188 / 0.82068

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？

答：

Layer (type)	Output Shape	Param #
batch_normalization_2 (Batch Normalization)	(None, 300)	1200
dense_1 (Dense)	(None, 256)	77056
dropout_1 (Dropout)	(None, 256)	0
dense_2 (Dense)	(None, 128)	32896
dropout_2 (Dropout)	(None, 128)	0
dense_3 (Dense)	(None, 64)	8256
dropout_3 (Dropout)	(None, 64)	0
dense_4 (Dense)	(None, 2)	130
Total params: 119,538		
Trainable params: 118,938		
Non-trainable params: 600		
None		

loss function: categorical_crossentropy, optimizer: adam

epoch: 20 (但只存 validation accuracy 最高的)

準確率：0.79597 / 0.79626

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。

答：

BOW: 兩者皆為 0.668。因為 BOW 的作法是將該句子所有單字的 embedding vector 加總後做平均，所以單純的語序變換不會造成結果的差異。

RNN: 前者：0.239，後者：0.647。因為 RNN 比起 BOW 較能抓取到句子的語序特徵，造成兩者的情緒分數不同。且後者分數較高，也與一般人的理解較相符（but 連接詞後的句子通常是說話者想強調的重點。）

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。

答：有標點：0.82625 / 0.82521，沒標點：0.82188 / 0.82068，有標點較好。推測是因為標點其實也是一項非常重要的特徵，比如說 "It is good!" 和 "It is good!???"，前項是正面情緒，後項可能帶有質疑的語氣而偏負面。因此有標點符號會造成準確率較好。

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。

答：先只用有 label 的 data 訓練 5 個 epoch，此時正確率為 0.82188 / 0.82068，接著用訓練好的模型對 non-label 的 data 進行 predict，並將預測出來情緒分數 > 0.8 的標記為 1，情緒分數 < 0.2 的標記為 0，再將這些 data 加進去原本有 label 的 data 一起訓練 5 個 epoch，正確率提昇為 0.82282 / 0.82154，有些微的提昇，可以理解成因為訓練集增大，所以 model 比較好訓練，準確率也變高。至於沒有提昇太多的原因推測是因為我將 threshold 訂成 0.8 和 0.2，而 model 對這些句子通常已經能很好的抓到特徵了（才會這麼肯定是正面或負面），所以準確率的增幅較有限。