

## Machine Learning HW6 Report

學號：b06902067 系級：資工二 姓名：許育銘

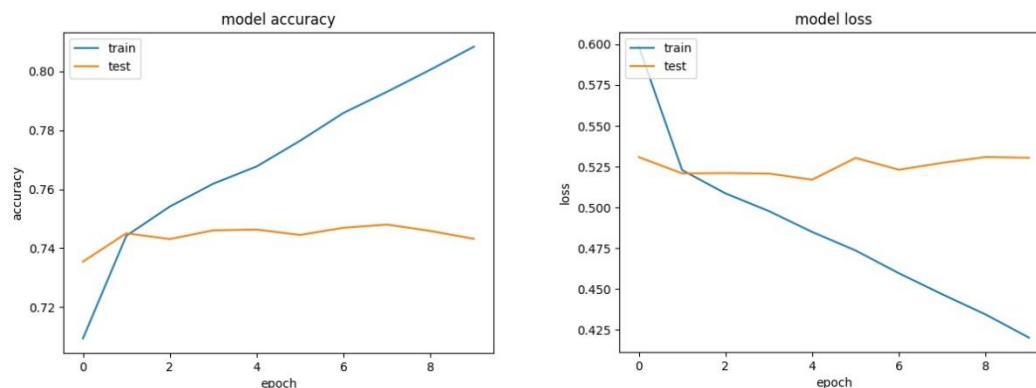
1. (1%) 請說明你實作之 RNN 模型架構及使用的 word embedding 方法，回報模型的正確率並繪出訓練曲線\*

RNN model:

Layer (type)	Output Shape	Param #
lstm_1 (LSTM)	(None, 100)	160400
batch_normalization_1 (Batch Normalization)	(None, 100)	400
dropout_1 (Dropout)	(None, 100)	0
dense_1 (Dense)	(None, 2)	202
Total params: 161,002		
Trainable params: 160,802		
Non-trainable params: 200		

word embedding：將所有留言分詞後用 Word2Vec 訓練

Word2Vec(size=300, iter=15, window=5, min\_count=3)



可以看出 validation loss 在 4、5 個 epochs 後有逐漸上升的趨勢，所以採用的是 5 個 epochs 後做 early stop 的結果。

public score:0.7427

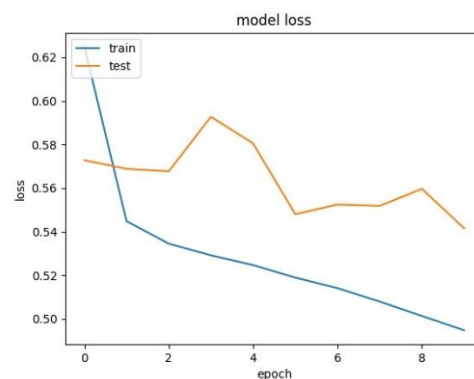
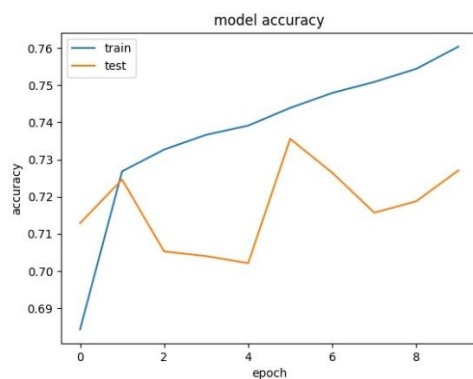
private score:0.7393

2. (1%) 請實作 BOW+DNN 模型，敘述你的模型架構，回報模型的正確率並繪出訓練曲線\*。

由於直接把每個詞都代表一個維度的話，我電腦的 RAM 負荷不了，所以我是把每個字用 word2vec 轉成向量後相加作為輸入。

DNN model:

Layer (type)	Output Shape	Param #
dense_1 (Dense)	(None, 200)	60200
dense_2 (Dense)	(None, 200)	40200
batch_normalization_1 (Batch Normalization)	(None, 200)	800
dropout_1 (Dropout)	(None, 200)	0
dense_3 (Dense)	(None, 100)	20100
dense_4 (Dense)	(None, 100)	10100
batch_normalization_2 (Batch Normalization)	(None, 100)	400
dropout_2 (Dropout)	(None, 100)	0
dense_5 (Dense)	(None, 2)	202
Total params: 132,002		
Trainable params: 131,402		
Non-trainable params: 600		



public score:0.7229

private score:0.7255

3. (1%) 請敘述你如何 improve performance (preprocess, embedding, 架構等)，並解釋為何這些做法可以使模型進步。

preprocess:我每個句子取的詞的各數為 100 個，當不足 100 個時，就重複那個句子直到補足 100 個，因為重複說一個句子幾遍不太會影響他是不是惡意留言，而且比較不會被句子長短影響

embedding:在 model 最前面加一層 embedding layer，把 word2vec 得到的向量再拿去 embedding layer 做修正，由於一個詞可能有不同意思，所以可能在得知這是否為惡意留言後會再改變詞之間的關係。

架構:兩層的 lstm 會比一層的好，但也更容易 overfit，我在 2~3 個 epochs 就會做 early stopping。

此外，我也訓練了三個 model 做 ensemble。

4. (1%) 請比較不做斷詞 (e.g., 以字為單位) 與有做斷詞，兩種方法實作出來的效果差異，並解釋為何有此差別。

	不做斷詞	有做斷詞
public score	0.7399	0.7427
private score	0.7355	0.7393

我認為是因為中文一個字有時無法表達明確的意思，必須要完整的詞才能判斷是否帶有惡意，比如說「白」一般當顏色時沒有貶意，但「白痴」卻是貶意字，這會讓 word embedding 的效果變差。

5. (1%) 請比較 RNN 與 BOW 兩種不同 model 對於 "在說別人白痴之前，先想想自己" 與 "在說別人之前先想想自己，白痴" 這兩句話的分數 (model output)，並討論造成差異的原因。

(註:此題所用的 model 與之前不同，之前的輸出是 0、1 的二維向量取 softmax，此題為一維向量取 sigmoid)

RNN:前者為 0.653，後者為 0.685

BOW:前後皆為 0.429

由於這兩個句子詞的組成完全相同，所以 BOW 的結果完全相同。

但 RNN 與句子的順序有關，所以「白痴」的位置會影響結果。