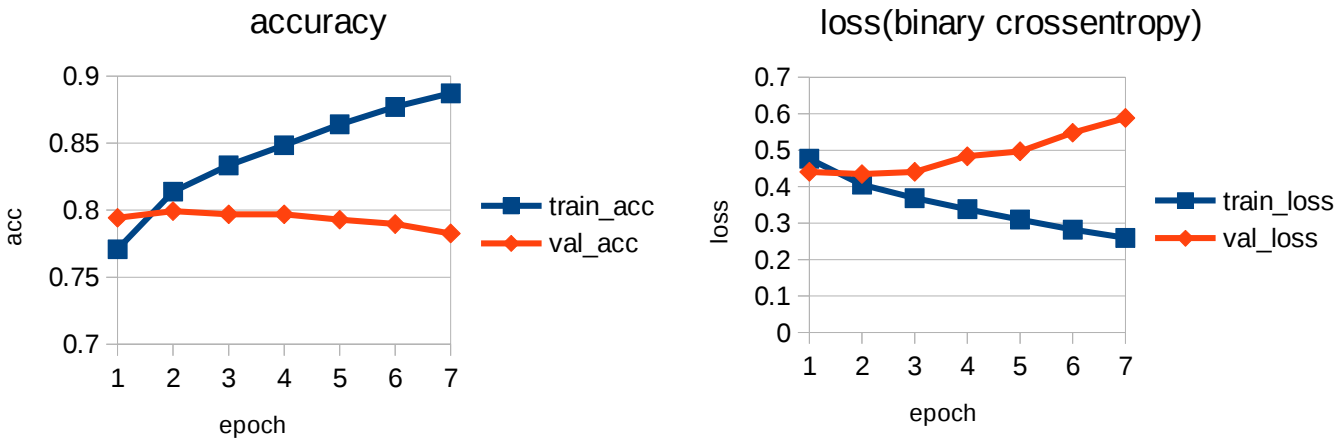
學號：B04901155 系級： 電機三 姓名：鄭閎

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：利用 keras 的 Tokenizer 和 Embedding 實做 RNN，模型架構如下：

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_1 (Embedding)      (None, 30, 128)           3840000
_________________________________________________________________
lstm_1 (LSTM)                (None, 64)                49408
_________________________________________________________________
dense_1 (Dense)              (None, 1)                 65
=================================================================
Total params: 3,889,473
Trainable params: 3,889,473
Non-trainable params: 0
_________________________________________________________________
```



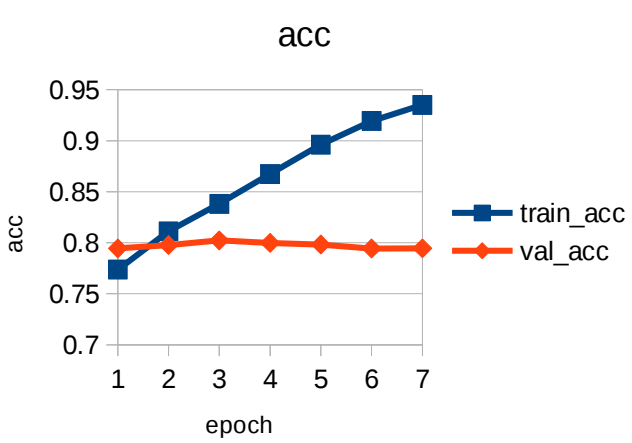
   epoch: 7 (earlystopping), optimizer: adam, loss: binary crossentropy
   左圖是訓練資料和驗證資料的準確率，右圖是訓練資料和驗證資料的 loss。其實經過第 3 個 epoch 之後 validation loss 就一直上升，所以才會 earlystopping。

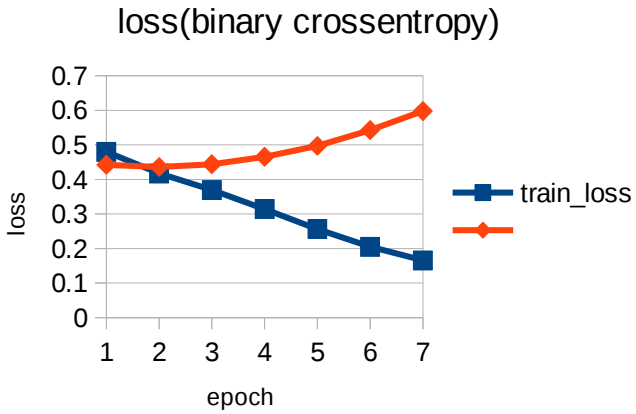
   最佳的 kaggle 分數：0.80097

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：使用 tokenizer.text_to_matrix 實現 BOW，模型架構如下：

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_1 (Dense)              (None, 512)               512512
_________________________________________________________________
dropout_1 (Dropout)          (None, 512)               0
_________________________________________________________________
dense_2 (Dense)              (None, 1)                 513
=================================================================
Total params: 513,025
Trainable params: 513,025
Non-trainable params: 0
_________________________________________________________________
```





epoch: 7 (earlystopping), optimizer: adam, loss: binary crossentropy
左圖是訓練資料和驗證資料的準確率，右圖是訓練資料和驗證資料的 loss。和 RNN 的情形類類似，經過第 3 個 epoch 之後 validation loss 就一直上升，同樣也有 earlystopping 的現象。

最佳的 kaggle 分數：0.80232

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：見下表。此分數是最後一層一個神經元的 dense 通過 sigmoid 產生，接近 1 代表偏正面，接近 0 是偏負面。

	第一句	第二句
Bag of word	0.52480	0.52480
RNN	0.39339	0.24116

Bad of word 不會考慮詞的順序，只考慮不同的詞出現的次數，因此對這兩句只是重排的例子產生的結果完全一樣。而 RNN 有記憶性，不同的排列順序對結果會有影響。

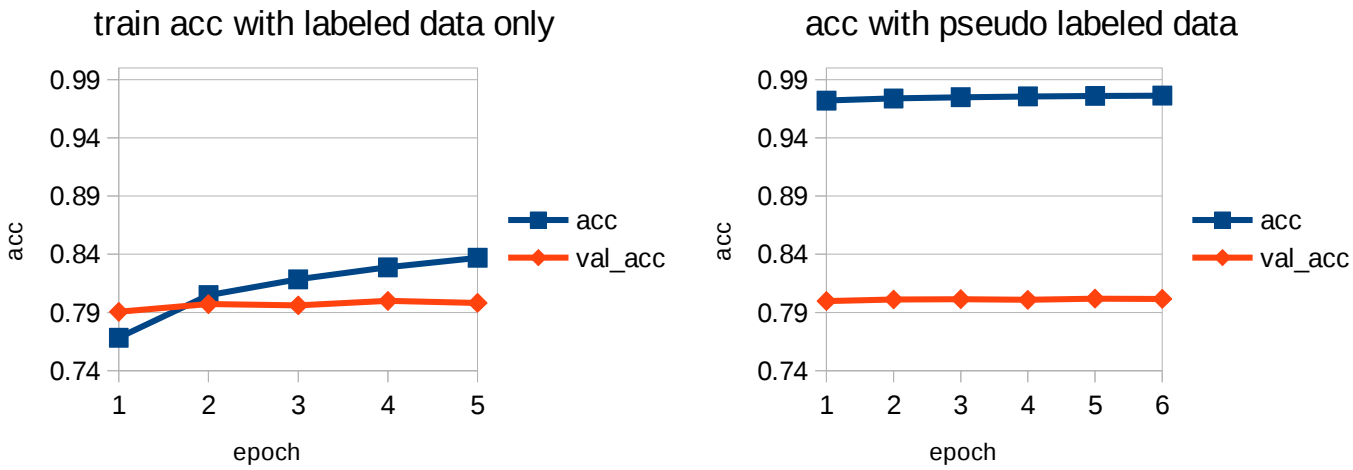
4. (1%) 請比較"有無"包含標點符號兩種不同 tokenize 的方式，並討論兩者對準確率的影響。
答：實做了 RNN 與 BOW，準確率如下：

	無包含標點符號	有包含標點符號
RNN	0.80097	0.79918
BOW	0.80232	0.79392

去掉標點符號對準確率會有提昇，但提昇的不多。推測原因是有一些像是逗點或是句點其實是沒有情緒的，但是出現很多次所以會進到字典裡面，導致影響結果。另一方面驚嘆號、問號等卻可能有反映情緒，去掉反而不好。因此這兩種因素影響下讓準確率和原本差不多，應該保留驚嘆號等可能影響情緒的表點符號而去除逗點等中性符號。

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：利用 training data 訓練出來的第一個 model 去預測所有沒標記的資料，算出的分數如果在 0.8 以上即給予 pseudo label 1，而小於 0.2 的資料標為 0。接著把這些資料加到 training data 裡面去重新訓練一次。
實做了 RNN，準確率如下：



左圖是一開始使用有標記的二十萬筆訓練資料訓練的過程（藍色為訓練資料的準確率），右圖是將一百多萬筆的無標記資料用一開始的模型預測後加入原本的二十萬筆再次訓練的過程。可以看到在訓練資料上的準確率非常高，原因是加入的這些資料其實是本來模型就會預測正確的資料，所以會有很高的準確率。

經過 semi-supervised training 後準確率由 0.79993 提昇到了 0.80195，並不是非常顯著。