學號：R06943153 系級： 電子碩一 姓名：蘇旻彥

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: 李明庭、游彥勝)
答：我最後使用的結果是由三個 model 作 ensemble 而成，以下 summary 敘述的是其中分數最高的 model_1，為單層的 Conv1d 加上單層的 LSTM，使用 Keras 內建的 Tokenizer，並且***沒有過濾標點符號***。中間使用的 Activation function 為 ***ReLU***，最後輸出層用的則是 ***sigmoid***。

```
Layer (type)                 Output Shape              Param #
=================================================================
embedding_1 (Embedding)      (None, 100, 128)          2560000
_________________________________________________________________
dropout_1 (Dropout)          (None, 100, 128)          0
_________________________________________________________________
conv1d_1 (Conv1D)            (None, 96, 64)            41024
_________________________________________________________________
max_pooling1d_1 (MaxPooling1 (None, 24, 64)            0
_________________________________________________________________
lstm_1 (LSTM)                (None, 70)                37800
_________________________________________________________________
dense_1 (Dense)              (None, 2)                 142
_________________________________________________________________
activation_1 (Activation)    (None, 2)                 0
=================================================================
Total params: 2,638,966
Trainable params: 2,638,966
Non-trainable params: 0
_________________________________________________________________
```

＊此 model 的詳細參數如下：

Token		Convolution	
num_words	40000	kernel_size	5
Embedding		filters	64
max_features	20000	pool_size	4
maxlen	100	Training	
embedding_size	128	batch_size	30
LSTM		epochs	2
lstm_output_size	70	Activation_function	relu

單 model_1 在 Kaggle_public 得到的分數是 0.81205，ensemble 之後是 0.81797。
接著稍微簡介另外兩個 model。Model_2 是使用自己***手刻的 dictionary***，最後輸出層使用的是 ***softmax***，其餘 model 主要架構及參數都和 model_1 相同。
Model_3 是***使用 testing_data 實作 semi-supervised***，詳細作法會在第 5 題說明，其餘所有架構和 model_1 相同。

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: 李明庭、游彥勝)
答：我所實作的 BOW model 以及參數如下：

```
Layer (type)                 Output Shape              Param #
=================================================================
dense_1 (Dense)              (None, 10000)             150010000
_________________________________________________________________
dense_2 (Dense)              (None, 5000)              50005000
_________________________________________________________________
dense_3 (Dense)              (None, 1000)              5001000
_________________________________________________________________
dense_4 (Dense)              (None, 100)               100100
_________________________________________________________________
dense_5 (Dense)              (None, 2)                 202
=================================================================
Total params: 205,116,302
Trainable params: 205,116,302
Non-trainable params: 0
_________________________________________________________________
```

Token : num_words = 15000, mode = count, filters=' ' (無過濾),
Batch_size = 128, Epochs=2
在 Kaggle_public 得到的分數是 ***0.79938***，相較 RNN 低很多。

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: 李明庭、游彥勝)
答：BOW 及 RNN(以第一題的 model_1 做比較)在兩句中分別得到的分數如下：

		BOW		RNN	
Class		0	1	0	1
Score	Sentence1	0.1508	0.8516	0.4629	0.5279
	Sentence2	0.1508	0.8516	0.0949	0.9074

可以看出由於 BOW 因為沒有單字順序的問題，這兩個句子對它來說都是一樣的，所以分數也都一樣。但是 RNN 可以看出單字順序的差異，雖然在第一個句子還是判斷錯誤，把它判定為正面，但是可以看出分數不是一面倒的導向正面；而第二個句子則是很明顯的判斷出是正面語氣。

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: 李明庭、游彥勝)
答：***有包含***標點符號的 tokenize 在 Kaggle_public 得到的分數是 ***0.81205***，而***不包含***的分數則是 ***0.80614***，相較之下輸了一點。我自己推測是因為標點符號在語意中有轉折或是停頓的語氣，所以有包含的可以學到這些東西，分數會高些。

5. (1%) 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。
   (Collaborators: 李明庭、游彥勝)
   答：在 model_3 中所實作的 semi-supervised 是***以 testing_data 作 nolabel_data***，以 model_1 預測之後，將預測分數***高於 0.9***(大約三分之一)的資料挑出，將預測結果做為 label 後加入 training_data。在 Kaggle_public 得到的分數是 ***0.81291***。