學號：R06921011 系級： 電機碩一 姓名：劉逸霖

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：

Layer (type)	Output Shape	Param #
embedding_1 (Embedding)	(None, 30, 128)	2560000
spatial_dropout1d_1 (Spatial	(None, 30, 128)	0
gru_1 (GRU)	(None, 64)	37056
dense_1 (Dense)	(None, 32)	2080
dropout_1 (Dropout)	(None, 32)	0
dense_2 (Dense)	(None, 1)	33

Total params: 2,599,169
Trainable params: 2,599,169
Non-trainable params: 0

用一層 GRU 接兩層 DNN，loss function 使用 crossentropy，並用 adam 作為 optimizer，用 2 個 epoch 就夠了，接下來就會 overfit 了。
Kaggle : (0.81151+0.80993)/2 = 0.81072

用 Bidirectional + 兩層 GRU + self-training 效果會比較好，不過依然沒有過 strong baseline，這邊就不探討細節了。

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 #
dense_1 (Dense)	(None, 1024)	3073024
dropout_1 (Dropout)	(None, 1024)	0
dense_2 (Dense)	(None, 512)	524800
dropout_2 (Dropout)	(None, 512)	0
dense_3 (Dense)	(None, 128)	65664
dropout_3 (Dropout)	(None, 128)	0
dense_4 (Dense)	(None, 32)	4128
dropout_4 (Dropout)	(None, 32)	0
dense_5 (Dense)	(None, 1)	33

Total params: 3,667,649
Trainable params: 3,667,649
Non-trainable params: 0

建一個 3000 個字的字典，把每一句話都用一個長度 3000 的 vector 表示，直接接 5 層 DNN，loss function 使用 crossentropy，並用 adam 作為 optimizer，用 2 個 epoch。
Kaggle : (0.79170+0.79266)/2 = 0.79218

BOW 的效果沒有 RNN 好。

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: 0.6438599 / 0.6438599
RNN: 0.53232384 / 0.84669816

BOW 兩句話預測結果一樣，是因為 BOW 是把一句話中出現的詞丟進一個袋子，並不管這些詞的前後順序，所以這兩句話會一樣。
對於 RNN 而言，會考慮一句話的前後順序，通常對我們而言，BUT 後面接的那個句子比較重要，所以那會影響情緒分數比較大，導致第二句的分數高於第一句，但兩者其實都還算是正面，都有大於 0.5。

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：
如果濾掉全部的標點符號，準確率為: (0.80322+0.80479)/2 = 0.80400
如果把空格跟換行濾掉，準確率為: (0.81151+0.80993)/2 = 0.81072

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。
答：

我直接使用 testing data 來做 semi-supervised，將 predict 結果中大於 0.88 及小於 0.12 的 data 拿出來，加上 label，加入 training data 一起，在不同 model 中效果不同，有的可以多 0.4%的準確率。