學號：R08946006　系級：資料科學學程碩一　姓名：周逸平

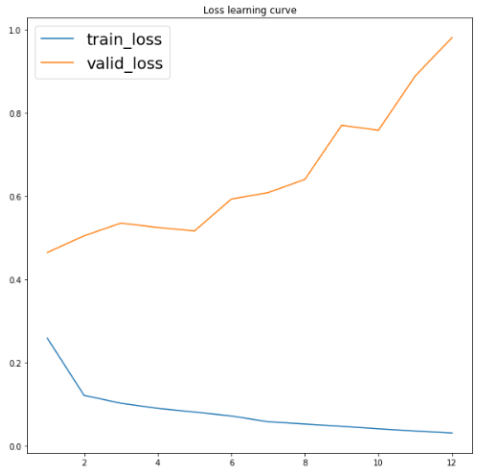
1. **(1%) 請說明你實作的 RNN 的模型架構、word embedding 方法、訓練過程 (learning curve)和準確率為何？ (盡量是過 public strong baseline 的 model)**

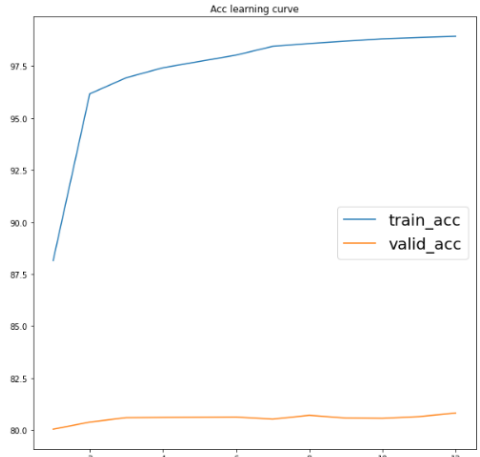
模型架構參見尾頁圖。

Embedding 方法選擇 skip-gram，embedding size = 300，sent_len 設定 30，理由請見 Question 3

使用 Adam with lr = 0.001

最終 kaggle 準確率為：0.82580





2. **(2%) 請比較 BOW+DNN 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"這兩句的分數(過 softmax 後的數值)，並討論造成差異的原因。**

sent1 = "today is a good day, but it is hot"

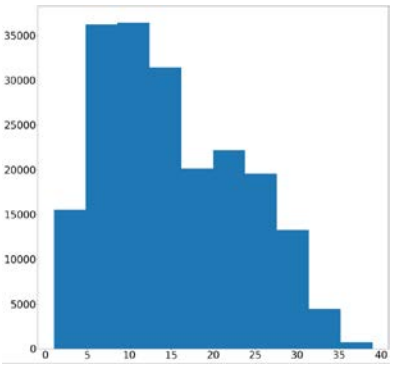
sent2 = "today is hot, but it is a good day"

	Sent1	Sent2
BOW+DNN	0.4202	0.3422
RNN (kaggle best)	0.0105	0.9995

以人的角度來看，可以發現第一句話強調的是「今天很熱」，而後者是「是個好日子」故代表順序會影響句子的語意表達，而使用 BOW+DNN 時只純粹計算句子中詞彙的出現次數，並不考慮順序以及文法等，相對的 RNN 配合 skip-qram 可以有效偵測句子內的組成，故 RNN 相對可以順利的分辨出兩句話的差異

3. **(1%) 請敘述你如何 improve performance（preprocess、embedding、架構等等），並解釋為何這些做法可以使模型進步，並列出準確率與 improve 前的差異。（semi supervised 的部分請在下題回答）**

原先只使用助教的 tutorial 一路跑到底，得到的成績為 kaggle : 0.82073。
接著我使用了以下方法增進準確率：



1. sen_len 選擇 30：我將所有句子繪製成 density plot 觀察多數句子長度為何，發現長度 30 可以包含 90%以上的句子長度。

2. embedding_size 選擇 300：用更高維的空間來解釋複雜的語句。

3.LSTM 啟用 bidirectional：讓 LSTM 可以從兩個方向觀察句子，讓模型更容易抓到語句中的特徵，如主受詞對應，語句順序等等。

4. 增加 hidden_dim 以及 num_layer：讓 LSTM 可以學習更複雜的語句結構。

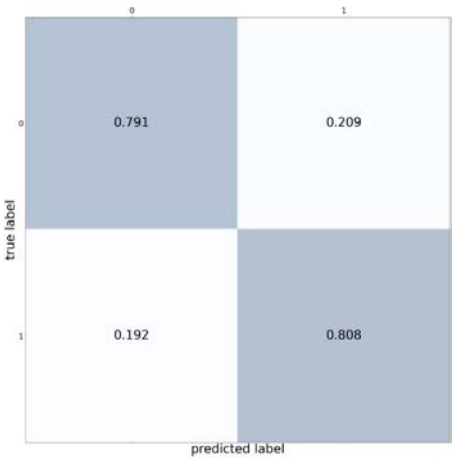
5. 啟用 LSTM 的 dropout：由於使用 2 ，使得 LSTM 有可能更偏激的學習 training dataset，故使用 dropout 來降低 overfitting 的問題

6. 增加 classifier 的層數：理由同 4

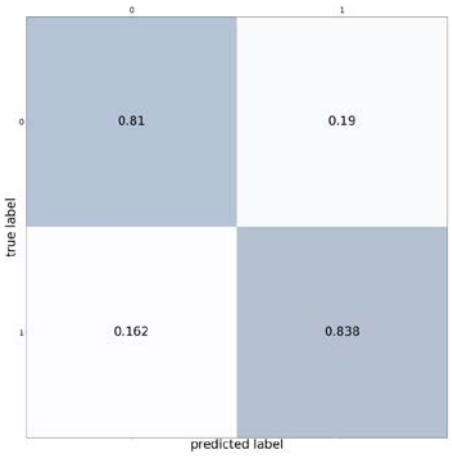
7. 啟用 classifier 的 dropout：理由同 5。

4. **(2%) 請描述你的 semi-supervised 方法是如何標記 label，並比較有無 semi-supervised training 對準確率的影響並試著探討原因（因為 semi-supervise learning 在 labeled training data 數量較少時，比較能夠發揮作用，所以在實作本題時，建議把有 label 的 training data 從 20 萬筆減少到 2 萬筆以下，在這樣的實驗設定下，比較容易觀察到 semi-supervise learning 所帶來的幫助）。**

先用一簡單 tuned 過的 LSTM，於 valid set 上確認準確度以及 confusion matrix，如圖。
可以發現 0 的準確度以及 1 的準確度都靠近 0.8，故我使用相同 model 預測 train_x_no_label 之後，若是分數 > 0.9 我便判定為 1，<0.1 判定為 0，其餘捨棄。
不使用 0.8 以及 0.2 是避免太鬆的 threshold 造成資料預測錯誤。



而後使用一較複雜的 LSTM 模型 (問題一的) 再一次 train，使用的資料集為 train[:180000] + train_no_label_picked，共 80w 句以上。

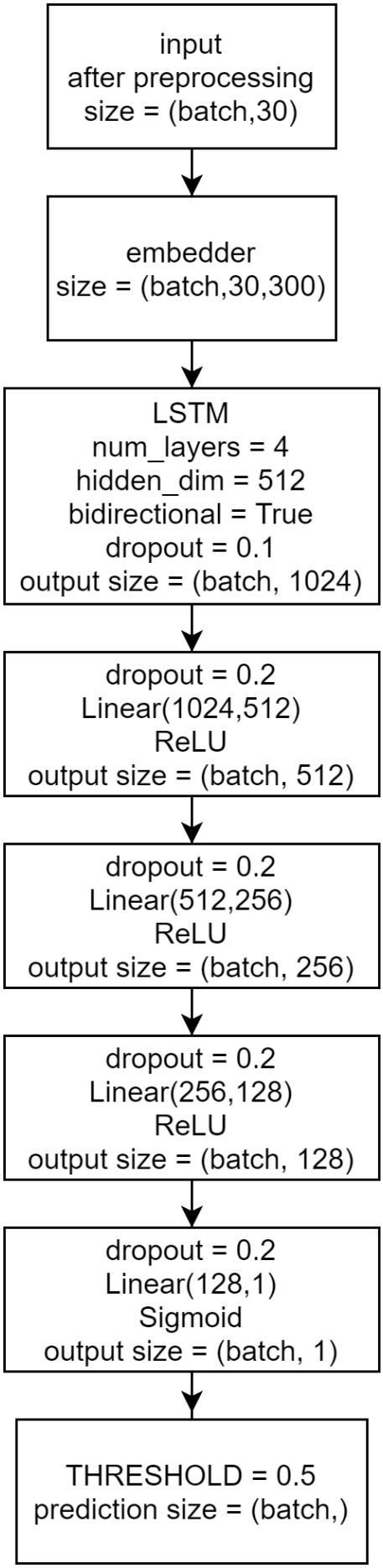


而後再一次使用複雜的 LSTM 模型做先前的步驟，得到右邊的 confusion matrix，再重複先前步驟。

最後得到的 kaggle 結果如下：

	Kaggle score
Tutorial	0.81773
Final model with first semi	0.82013
Final model with sec semi	0.82580

LSTM 架構圖


input
after preprocessing
size = (batch,30)
embedder
size = (batch,30,300)
LSTM
num_layers = 4
hidden_dim = 512
bidirectional = True
dropout = 0.1
output size = (batch, 1024)
dropout = 0.2
Linear(1024,512)
ReLU
output size = (batch, 512)
dropout = 0.2
Linear(512,256)
ReLU
output size = (batch, 256)
dropout = 0.2
Linear(256,128)
ReLU
output size = (batch, 128)
dropout = 0.2
Linear(128,1)
Sigmoid
output size = (batch, 1)
THRESHOLD = 0.5
prediction size = (batch,)