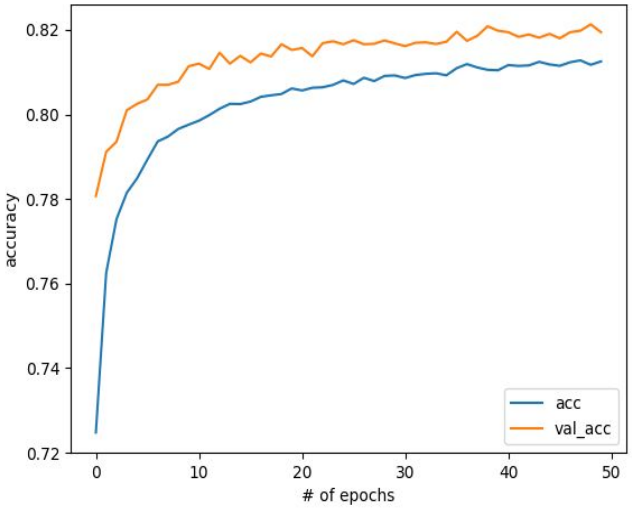
學號：B04902105 系級： 資工三 姓名：戴培倫

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？

我先將labeled_data, unlabeled_data, testing_data用gensim的Word2Vec pretrain出一個將55776個最常用的字投到100維的mapping，將每個句子長度補到39個字，map到100維的空間，再丟進三層LSTM ⇒ 兩層Dense。

Layer (type)	Output Shape	Param #
lstm_1 (LSTM)	(None, 39, 64)	42240
lstm_2 (LSTM)	(None, 39, 64)	33024
lstm_3 (LSTM)	(None, 64)	33024
dense_1 (Dense)	(None, 64)	4160
dense_2 (Dense)	(None, 1)	65

Total params: 112,513
Trainable params: 112,513
Non-trainable params: 0



準確率在5.6個epoch時就超過80%，且一直在上升，val_acc一直都高於acc大約1%，kaggle public score為0.82040。

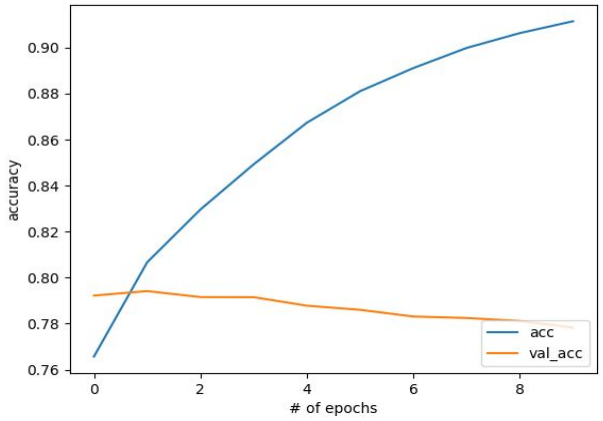
因為accuracy一直上升，也未見overfitting的情況，我有將model train到100個epoch，期間準確率緩緩上升，kaggle public score為0.82286。兩個ensemble的結果為0.82397。

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？

Layer (type)	Output Shape	Param #
dense_1 (Dense)	(None, 100)	764400
dropout_1 (Dropout)	(None, 100)	0
dense_2 (Dense)	(None, 64)	6464
dropout_2 (Dropout)	(None, 64)	0
dense_3 (Dense)	(None, 64)	4160
dropout_3 (Dropout)	(None, 64)	0
dense_4 (Dense)	(None, 1)	65

Total params: 775,089
Trainable params: 775,089
Non-trainable params: 0

我從labeled_data, unlabeled_data, testing_data中找出出現超過30次的單字，共7643個，將每個句子轉為7644維的one-hot vector，丟進四層Dense layer，其間皆Dropout 0.4。



從準確率可見在一個epoch之後就開始overfit，val_acc也開始下降。調整過nn層數與dropout rate，但都還是很容易overfit。推測是因為很分散的one-hot vector很容易完美地fit到結果。

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。

因為這兩句的單字組成一模一樣，因此用bag of word model判斷這兩句的情緒分數相同，皆為0.5426169，判斷為正面。推測是因為good明確代表正面，而hot較為模糊。

而用RNN model predict出的結果就有很顯著的差異，

第一句：0.45997903 ⇒ 負面（模糊）

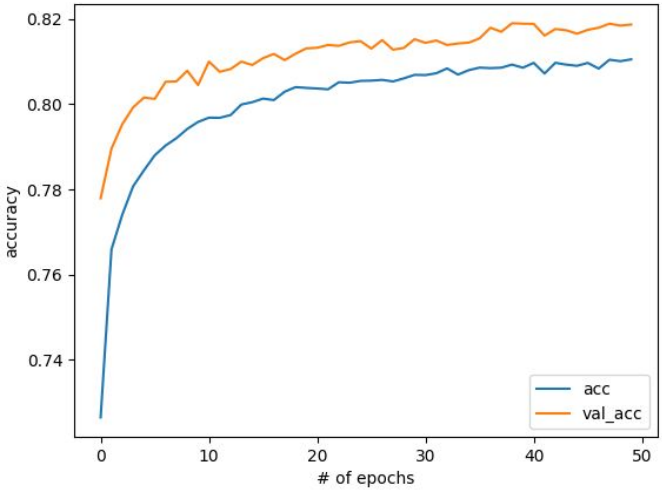
第二句：0.90783632 ⇒ 正面（明確）

而從語句本身的語意分析，發現判斷的結果以及模糊程度是很合理的。接在but後面的句子是重點。因此第一句重點為”it is hot”，第二句重點為”it is a good day”。good基本上代表的就是正面，而hot為稍微傾向負面的情緒形容詞。

第一句：前面正面，後面負面（重點）⇒ 稍微傾向負面

第二句：前面負面，後面正面（重點）⇒ 明確傾向正面

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。



將標點符號去除後，準確率略差一點，但無明顯差異。推測是因為標點符號能多少表達情緒，故去除後準確率略為下降，但標點符號在字典中比例太少(137 / 55776)，對結果影響有限。

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。

我將unlabeled_data用原本model預測，以放進training set的標準做了兩種觀察：

output > 0.8 or output < 0.2:

acc沒有比較好，但val_acc非常高。觀察增加的data後發現output>0.8的有四十萬筆，而output<0.2的只有七萬筆，推測是因為資料比例過於懸殊，而預測結果為正面即可有極高正確率。

output > 0.9 or output < 0.1:

output > 0.9的有86829筆，output < 0.1的有57334筆，比例比較接近，發現正確率有一些進步，上升速率也比較快，val_acc甚至接近83%。