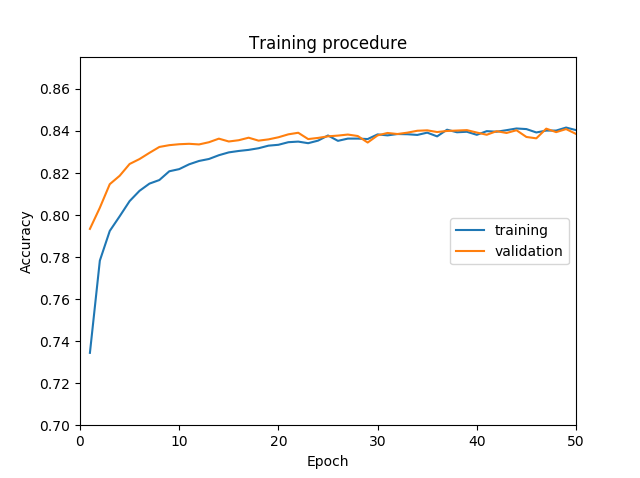
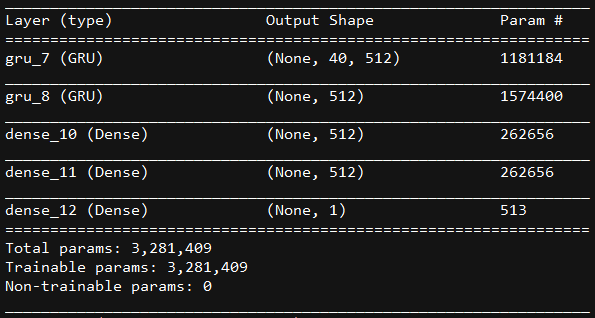
學號：b04901060 系級：電機三 姓名：黃文璁

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: 無) 答：**



此為單一模型中結果最好的，首先大致對文字資料進行下列預處理：

1. 只保留 “.?!” 三種標點符號和英文、數字。

2. 處理疊字問題。由於文字來源是Twitter故有許多疊字出現。

3. 處理基本的stemming和詞頻低於一定值的詞。

預處理後利用gensim的**word2vec**進行word embedding，維度為256。

**RNN**架構為： **2x GRU**: units=512, dropout=0.5, recurrent\_dropout=0.5

**2x Dense**: units=512, activation=selu

optimizer=adam, loss=binary\_crossentropy

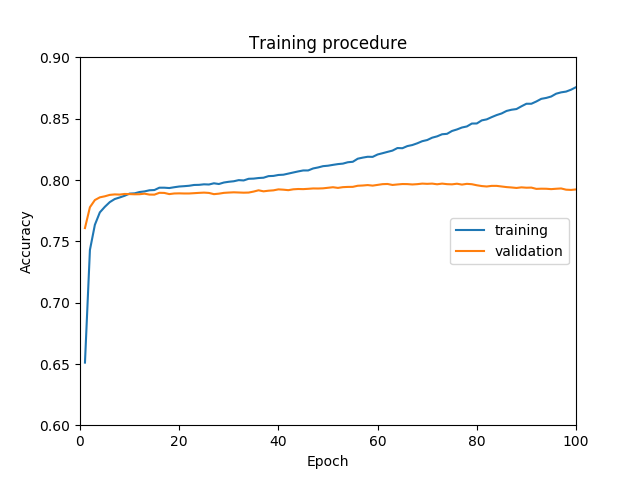
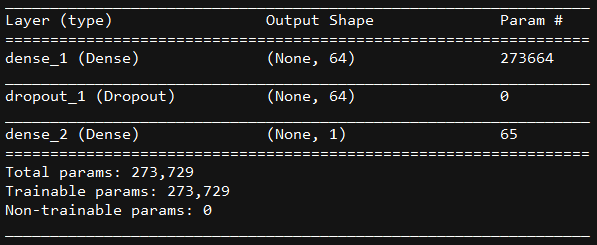
由上圖，training 和validation accuracy大概都在25個epoch時收斂到約0.835。

　　Kaggle上準確率為： **Public**: 0.83533 **Private**: 0.83348 | **平均**：0.83441

此外還實作了10個模型的ensemble，其他模型和上述的模型參數有些許不同。

　　Kaggle上準確率為： **Public**: 0.83947 **Private**: 0.83840 | **平均**：0.83894

1. **(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: 無) 答：**



預處理大致同RNN，另外還處理了skipwords，BOW的維度約為4000。

**BOW**架構為： **Dense**: units=64, activation=relu

**Dropout**: rate=0.2

optimizer=adam, loss=binary\_crossentropy

由上圖，validation accuracy大概在5個epoch時收斂到約0.79。

另外由圖中可以觀察到和RNN相比，BOW的overfit情況更明顯。

　　本機測試1/10資料validation的準確率為：**0.79235**

1. **(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: 無) 答：**

　　就我們看來，第一個句子稍偏負面，而第二個句子則較明確為正面。

**Prediction** **Label**

使用**RNN**來預測兩句話，得到的結果為： 第一句： 0.165759 0

第二句： 0.992874 1

使用**BOW**來預測兩句話，得到的結果為： 第一句： 0.680686 1

第二句： 0.680686 1

　　可以觀察到RNN對這兩句話的情緒都較為肯定（預測結果很接近0或1）

而BOW則顯然無法分辨這兩句話的情緒差異，這是由於兩個句子中的詞頻率相同。

對RNN來說，會考慮到前後關係，故應能根據 ‘but’ 的語氣轉折預測出正確的結果。

而BOW則無法判斷前後文，故語氣有轉折的句子就有可能判斷錯誤。

1. **(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: 無) 答：**

使用的Model同第一題。分別考慮標點符號的有無：

**有**標點符號： **Public**: 0.83406 **Private**: 0.83261 | **平均**：0.83333

**無**標點符號： **Public**: 0.82390 **Private**: 0.82390 | **平均**：0.82390

這是由於某些標點符號對文字的情緒有決定性的影響，最明顯的為 ! 和 ? 兩種。

　　例如有些句子去掉標點後會幾乎看不出情緒成分，但若加上驚嘆號的話就可以較明顯的看出情緒。而問號則關係到句子是否為疑問句，也可能對預測產生影響。除了這兩種外，刪節號 … 等符號也可能有一些影響。

1. **(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: 無) 答：**

從unlabeled data中取出40萬筆和training data不重複的句子，先利用Kaggle上準確率為0.83349 / 0.83193 (平均 0.83271) 的模型對這40萬筆資料做預測得到prediction，再保留預測值 >0.95 / <0.05的句子，分別標上pseudo label 1 / 0，再將這些句子和原本的20萬筆training data一起訓練和原本相同的模型。

　　經過上述處理後，semi-supervised training時大概共有47萬筆資料，訓練後在Kaggle上的準確率為0.83499 / 0.83411 (平均0.83455)。平均下來大概進步了0.2%。