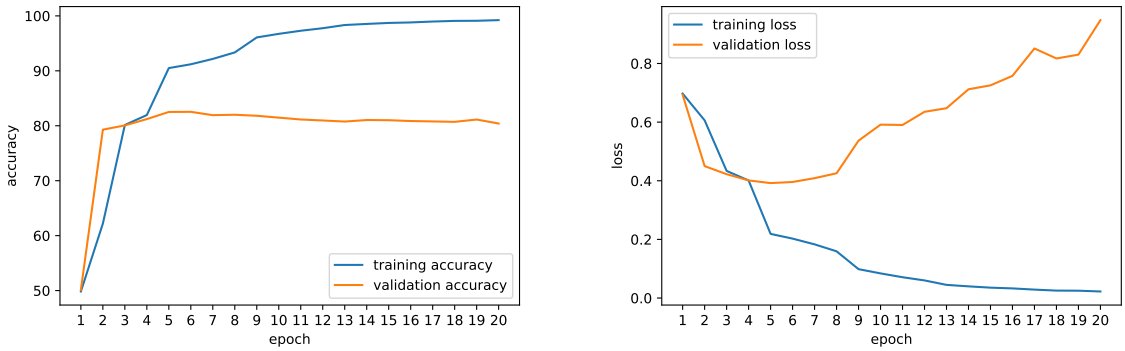
Machine Learning 2020 - Homework 4 Report

學號：b08902100, 系級：資工一, 姓名：江昱勳

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？

   我的 RNN 模型架構由三層雙向的 LSTM 加上三層的全連接層組成，每層 LSTM 後接帶有機率 0.5 的 Dropout 層，而每層全連階層之間的活化函數則是使用 ReLU，而最後會將輸出通過 Sigmoid 函數讓數值在 0 1 之間。word embedding 的部份使用 gensim 內建的 word to vector 模型做訓練，嘗試將文字輸出成 500 維的向量，使用 skip-gram 方式並且將 window 大小設定成 5，而一個字要出現超過 5 次才會將其考慮進來，最後訓練 25 個 epoch。

   RNN 的訓練過程可以參考下面的圖片



   最後我是將 validation accuracy 最高的 model 存下來使用，其準確率 (kaggle 上的 public score) 為 0.82232。

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。

   我建立了一個簡單的 DNN，總共由 5 層全連接層組成，除了最後一層以外，接以 ReLU 作為活化函數連接，最後一層則透過 sigmoid 函數將數值控制在 0 1 之間，而 BOW 的部份我則僅統計總出現次數超過 10 次的單字，並刪除一些標點符號以及停用詞來讓構成的向量大小不會太大；該 model 在 kaggle 上的 public score 為 0.79189。

   以 BOW+DNN 模型預測 “today is a good day, but it is hot” 與 “today is hot, but it is a good day” 的分數為 0.76685613，而 RNN 則輸出 0.2985。可以發現 RNN 將該句話視為叫為負面的語句，而 BOW+DNN 則相反，推測可能是因為 BOW+DNN 沒有整段話的資訊，無法分辨出 but 所代表的轉折語氣，而 RNN 因為會完整有順序的將一個句子閱讀完，因此叫 BOW+DNN 能體現語句中的轉折。

3. 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。

   我有參考過 “Pedro M. Sosa, (2017, June 7), *Twitter Sentiment Analysis using combined LSTM-CNN Models* [Online]. Available: `https://www.academia.edu/35947062/Twitter_Sentiment_Analysis_using_combined_LSTM-CNN_Models`” 描述的方法，將 LSTM 搭配 CNN 構築 model，然而我並沒有看到模型的進步。我也嘗試將 LSTM 改成 GRU 過，其 performance 與 LSTM 無太大的差異 (在 kaggle 上的 public score 都是 0.82232)。而嘗試換過 word to vector 的維度以及訓練次數皆沒有顯著的進步。

4. 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。

   我的 best model 有使用 self training 來加強他的 test data，而我的 semi-supervised 方法是自己挑選一組參數 k, r，會在每 k 個 epoch 後，利用當前的 model predict 那些 unlabeled data，並且將依照 r 的比例將較接近 1 與較接近 0 的資料挑除來，放入 training set 裡面，繼續訓練模型。我認為有為使用上述的 semi-supervised 方法訓練對於準確度可以有小小的增幅，且使用後可以稍微加快模型的收斂速度。

   我依照建議嘗試分別使用 self training 與不使用 self training 訓練了兩個模型，有使用的其在 kaggle 上的 public score 為 0.78970，而沒使用的則為 0.77847，從下列的圖片也可看出兩者訓練時的圖形其實差異不大。

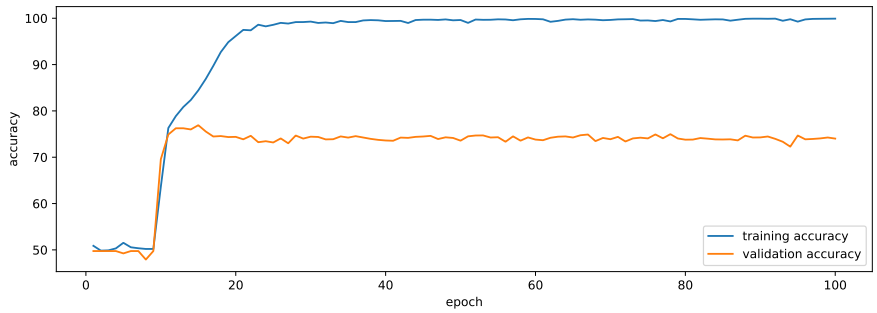


Figure 1: accuracy without self training

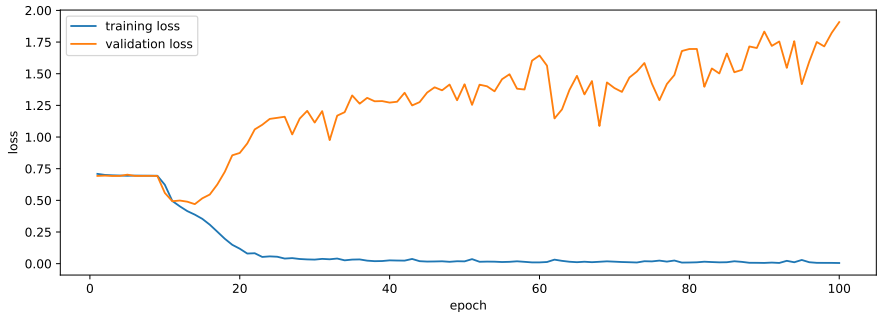


Figure 2: loss without self training

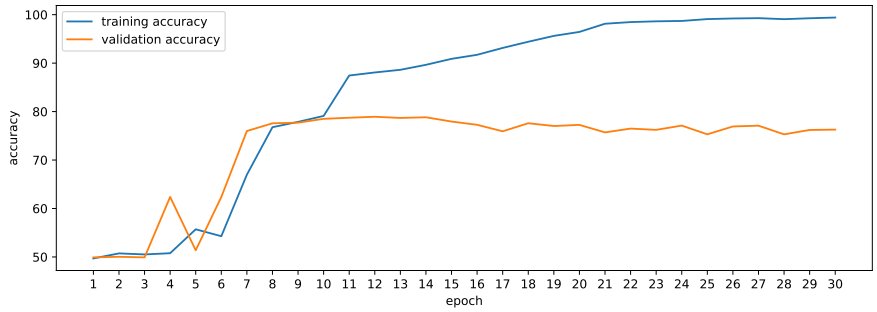


Figure 3: accuracy with self training

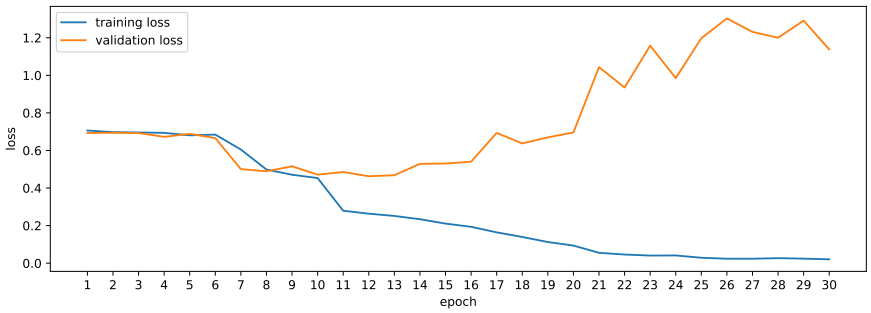


Figure 4: loss without self training