Machine Learning HW6 Report

學號：B04501073 系級：土木四　姓名：李利元

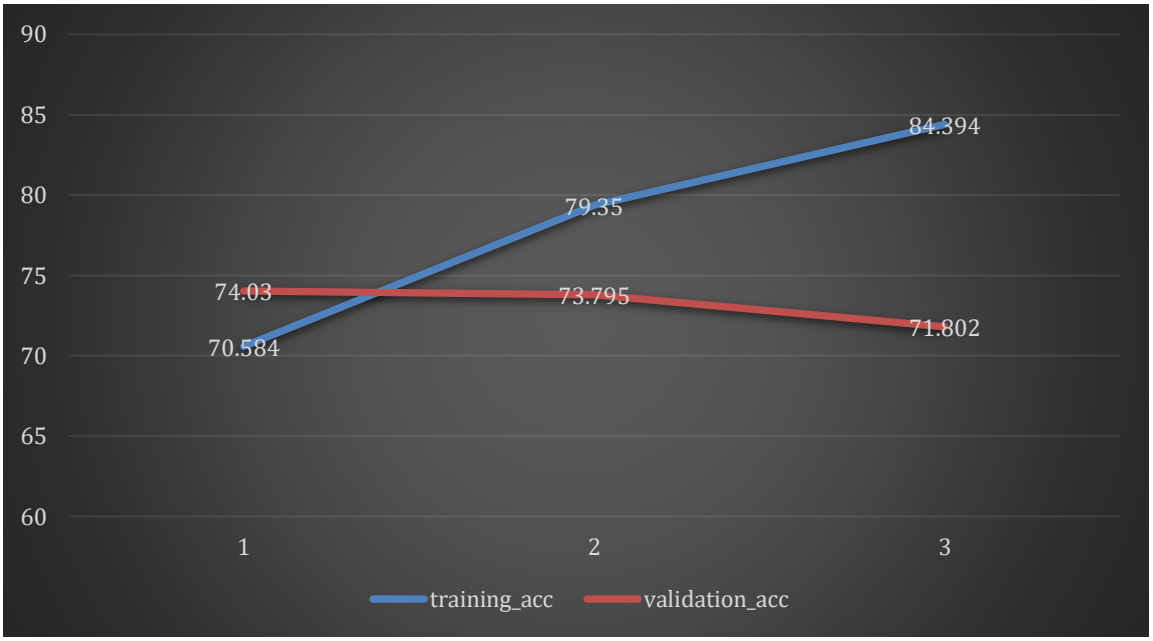
1. (1%) 請說明你實作之 RNN 模型架構及使用的 word embedding 方法，回報模型的正確率並繪出訓練曲線*

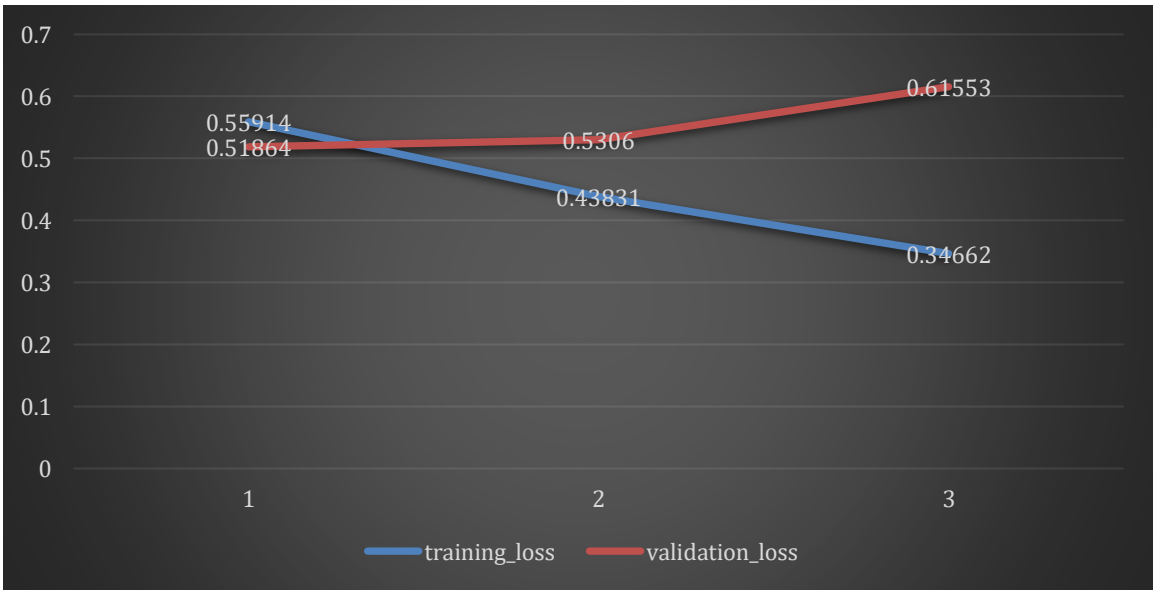
使用的模型架構為一層 LSTM 加上一層 DNN，word embedding 方法為 gensim.models 的 Word2Vec，iter 為 16，訓練過程曲線如下圖：

最後正確率為 Public:0.7154， Private:0.7186，不慎理想，推斷可能是因為資料前處裡出了一些問題。

2. (1%) 請實作 BOW+DNN 模型，敘述你的模型架構，回報模型的正確率並繪出訓練曲線*。

模型架構為一層 embedding(40810 維)加上三層 dnn，訓練過程如下：





最後正確率為 Public:0.73170， Private:0.72680，比 lstm 稍高一些，可能是因為這次的 data 並不太需要處理語序問題，故分數較高。

3. (1%) 請敘述你如何 improve performance（preprocess, embedding, 架構等），並解釋為何這些做法可以使模型進步。

   有試過在 Preprocess 過程中把標點符號以及表情符號刪掉，這個方法可以有效刪除對於判讀影響力較低的詞彙，讓 padding 的過程可以盡量把較有影響力的詞彙保留下來，另外增加 Word2Vec 的 iter 次數也可以讓 embedding 的 vector 更準確，進而讓預測率也提高

4. (1%) 請比較不做斷詞 (e.g., 以字為單位) 與有做斷詞，兩種方法實作出來的效果差異，並解釋為何有此差別。

   若無使用 jieba 套件做斷詞，並使用 LSTM，最後準確率為 Public:0.4873，Private:0.4828 基本上與亂猜的機率差不多，相比做斷詞之成功率 Public:0.7154，Private:0.7186 相差甚遠，此差異最大的原因就是語意問題，某些固定用法的成語或專有名詞可能都是由多個字組成，而我們人類在解析這些語句的意思時，也會把他們當作一體來解讀而不會分成好幾個部分，此外，若沒有使用斷詞，在 padding 大小相同的情況下，同樣的一組 data 可以存取的語意可能會大大的減少，因為每個字的意思都是需要被解讀的，因此可能會造成重要訊息消失的問題。

5. (1%) 請比較 RNN 與 BOW 兩種不同 model 對於 "在說別人白痴之前，先想想自己"與"在說別人之前先想想自己，白痴" 這兩句話的分數（model output），並討論造成差異的原因。

   RNN 結果：
   第一句話的分數為 0.5968，第二句話的分數為 0.6031，雖然未能成功判斷第一句話為非惡意言論，但可以觀察到它的分數確實比惡意言論稍低一些，可以知道說在 LSTM 的模型架構下，不同的 sequence 的確會造成不同的輸出值。

   BOW：
   兩個值分別為 0.6241、0.6165，第一句話甚至更接近惡意言論，可能是因為斷詞關係讓 BOW 的結有些許不同，但還是很明顯地看出 BOW 的架構無法處理順序上不同而造成的歧異性。