Machine Learning HW6 Report

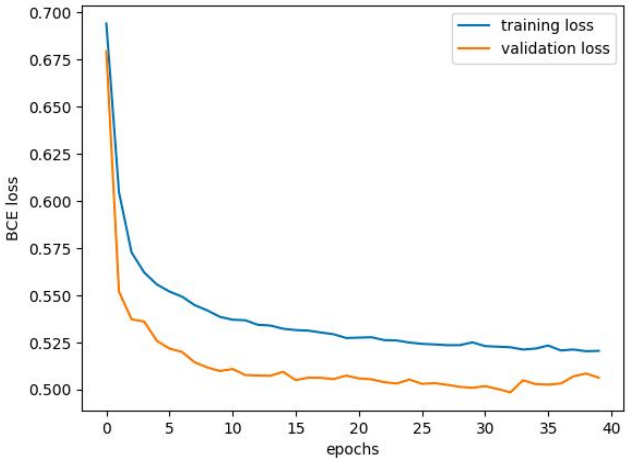
學號：B05901111　系級：電機三　　姓名：陳建成

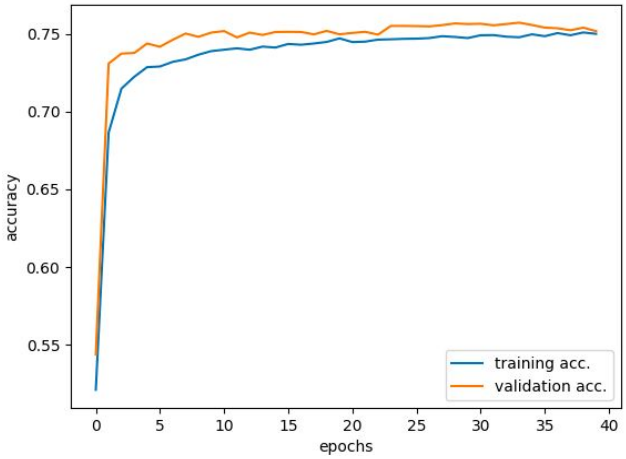
1. (1%) 請說明你實作之 RNN 模型架構及使用的 word embedding 方法，回報模型的正確率並繪出訓練曲線*

我的RNN模型是用GRU作為RNN unit，疊一層100個GRU的recurrent layer，後面跟兩層分別有50、25個neurons的fully connected layers，然後dropout都設0.5（包含GRU的recurrent dropout也是）。並且train 40個epochs。（其實前面主要嘗試LSTM但是明顯GRU比較快收斂）

Word embedding的部分用Gensim Word2Vec套件，embedding維度設為 200 維，minimum count設預設的 3 次，因而得到 45700 種 tokens（另外要再加上<UNK>和<PAD>共45702種），embedding training的window也是預設的3；對於不同長度的句子都padding（實為截長補短）到 40 個向量。

Kaggle results （正確率）：　private score: 0.74880, public score: 0.75530



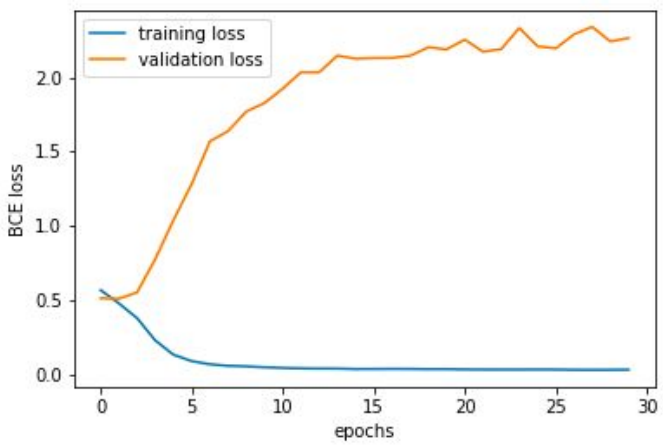


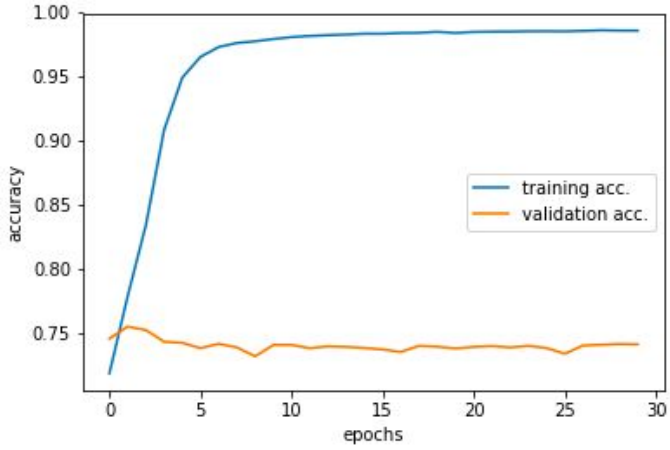
（實際上在Kaggle上的為四個模型ensemble的結果。）

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*。

我的BOW+DNN模型為簡單Dense的（400, 50, 25, 1）DNN，中間的activation function都是ReLU而最後一層放sigmoid，和前者一樣dropout設為0.5。因為BOW比較吃空間，因此我的minimum count設定到30才能成功訓練。至於<UNK>就不留給它一個維度了（這裡也不需要<PAD>）。

Kaggle results （正確率）：　private score: 0.74790, public score: 0.74460





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。

   增加embedding的維度以增加embedding可以代表的額外維度，架構上則是除了增加RNN unit（LSTM或GRU）的數量外，還有在後面fully connected layer的層數與neuron數量。增加embedding維度可以使其攜帶更多單詞的資訊，而Dense的層數與neuron數量可以增加model的capacity，但是會提高overfitting的風險。（不過因為實際上做LSTM training的時候還沒做到overfitting的程度所以目前影響不大～可能還有其他嘗試空間）

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。

   比較同樣的另一個以LSTM實作的model，其有做斷詞跟沒做的正確率：

   有做斷詞：

   Kaggle results （正確率）： private score: 0.73220, public score: 0.73790

   沒做斷詞：

   Kaggle results （正確率）： private score: 0.72720, public score: 0.73490

   由此可以看出，斷詞對於model判斷語意確實有幫助，因為可以讓模型了解真正語意的界限在哪裡。

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。

   比較RNN和BOW分別對兩句話的判斷結果，RNN對於兩句話的model output分別為0.42526013與0.5327055，但是BOW的判斷結果卻都是0.6019917。因此只有RNN才能正確將第一句依照人的想法判斷為非惡性留言。此現象的理由其實很簡單，因為BOW只能抓到出現的關鍵字而沒有順序之分，但是在這個句子中詞語的順序卻明顯的影響了語意。