

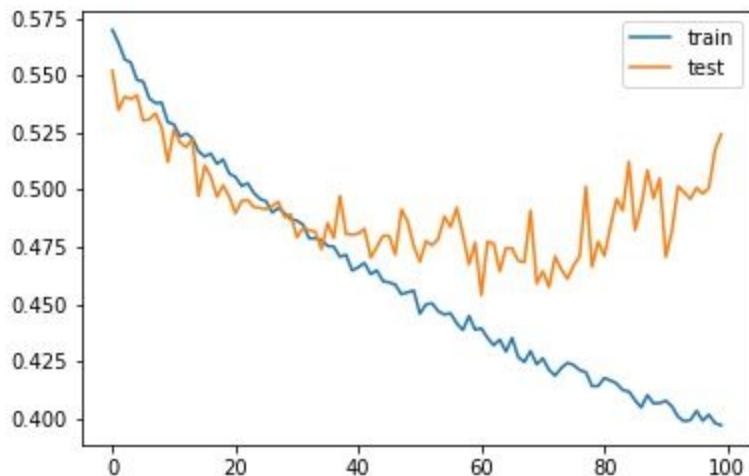
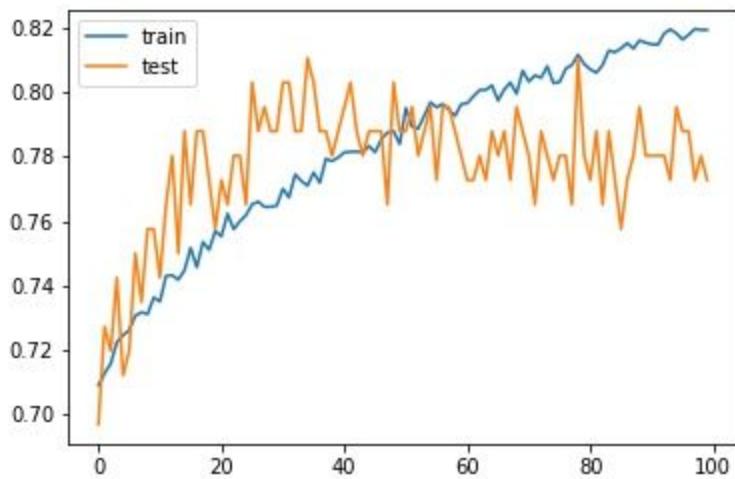
Machine Learning HW5 Report

學號：b04303006 系級：經濟四 姓名：劉傳筠

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*

本次使用Bi-Directional GRU模型兩層(hidden units = 30, dropout rate = 0.4, recurrent dropout = 0.4), epoch - 200。word embedding使用word2vec套件(min_count = 1, epoch = 20)。

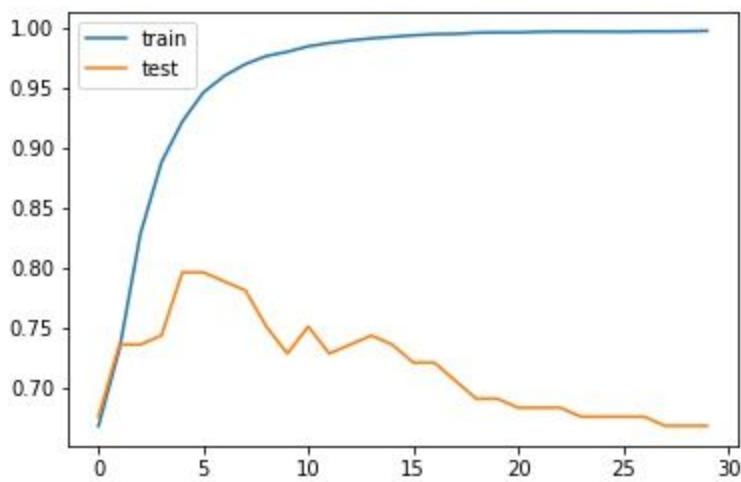
F1 - score: private - 0.79534, public - 0.79767



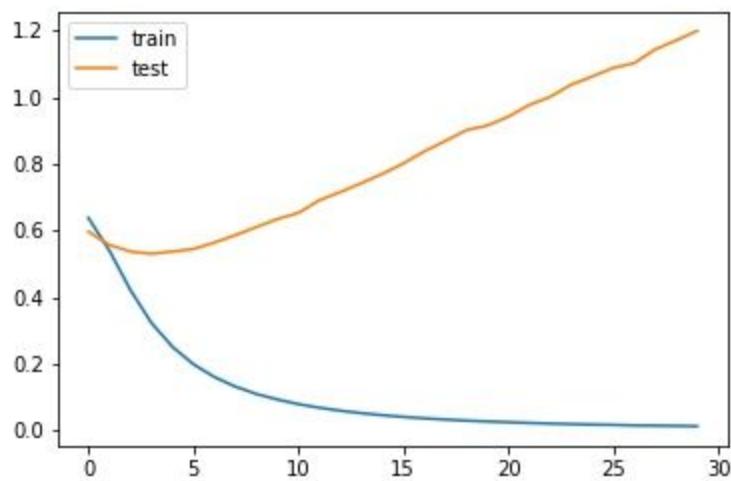
2. (1%) 請實作 BOW+DNN 模型，敘述你的模型架構，回報模型的正確率並繪出訓練曲線*。

BOW簡單使用空白來做斷詞。因為維度之間感覺也沒有什麼相關性，只是純粹的頻率，所以不想用太複雜的DNN模型，就疊一層10個units然後通過sigmoid function。這裡和上一題一樣使用隨機抽取的1%資料作為validation data。可以觀察到training set的accuracy上升的很快，最後上升到將近1.0，training loss也是一直降到接近0；但testing的再不到50個epoch就下降到0.7以下，testing loss也很高。

- Accuracy



- Loss



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。

其實沒有做什麼preprocessing, 主要是embedding有嘗試忽略出現較少次(min_count=3) , 但效果沒有很好。原因大概是其實我們的corpus也不是很大，如果要出現三次以上的話會丟失很多資訊。

這次主要比較有顯著的進步是在架構的調整。一開始嘗試一層GRU和LSTM都不能過simple baseline, 後來決定疊兩層，GRU就有過了。礙於時間因素，一開始都是train比較短的時間(epoch ≈ 100)，後來嘗試拉長成epoch=200就過了public的strong baseline(0.77906)。至於為什麼GRU表現比較好...我目前能給出的理由是因為GRU比較快，所以我有比較多時間調參數XD

最終是ensemble三個表現最好的模型，取得public=0.8000的結果。三個模型分別是：

- (1) 2 layer - GRU (min_count=1)
- (2) 2 layer - Bi-GRU (min_count=1)
- (3) 2 layer - Bi-GRU (min_count=3)

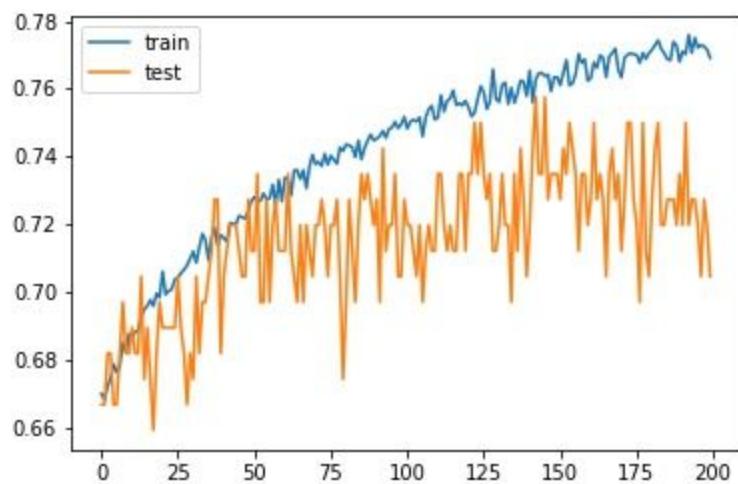
4. (1%) 請比較不做斷詞 (e.g.,用空白分開) 與有做斷詞，兩種方法實作出來的效果差異，並解釋為何有此差別。

用空白分開的分詞法需要train比較久，如圖所示，train了200個epoch都還沒有到0.8，testing accuracy也是跳來跳去，後期的成長不明顯。

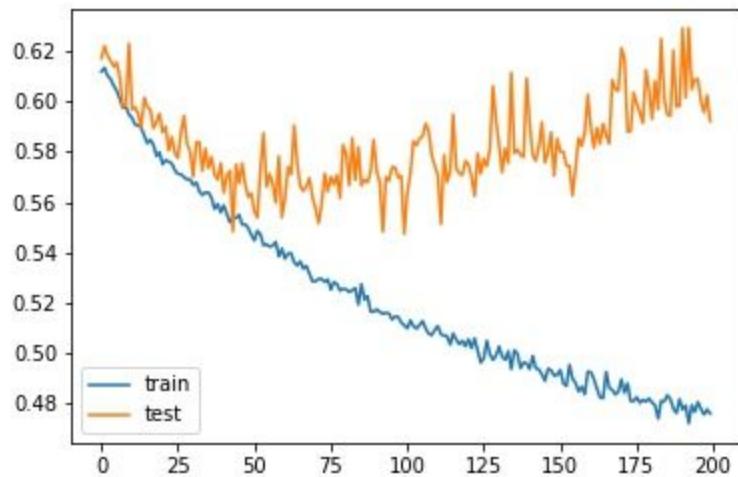
用spacy做斷詞時它會把具有獨立意義的字分開。例如縮寫It's 會被分成It & 's 這樣把It 和is分開，而不是把It's當作一個獨立的新的字。應該是因為spacy斷詞分的比較精確，才能在較短的時間內達到較高的testing & training accuracy

```
[ '@user', '@user', '@user', 'Means', 'nothing', '!', '!', 'Your', 'always', 'so', 'dramatic', 'and', 'none', 'of',
'your', 'promises', 'ever', 'come', 'true', 'not', 'sure', 'why', 'I', 'still', 'follow', 'you', '.', 'I', "m",
[starting', 'to', 'realize', 'you', 'are', 'full', 'of', 'crap', '.']
['@user', '.', 'It', 's', 'only', 'sexual', 'assault', 'when', 'the', 'other', 'does', 'not', 'agree', 'with', 'w
hat', 'he', 'is', 'feeling', '.', '"', 'He', 'said', 'before', 'picking', 'up', 'the', 'pace', 'his', 'foot',
'turning', 'and', 'pushing', 'against', 'his', 'cock', 'while', 'his', 'big', 'toe', 'and', 'other', 'toe', 'lift
', 'up', 'the', 'glands', '.', '.', 'Mm', '.']
['@user', '@user', '@user', 'what', 's', 'your', 'issue', 'with', 'antifa', '?']
['@user', 'Shame', 'on', 'the', 'tennis', 'hierarchy', 'who', 'stole', 'that', 'from', 'Serena', '.', 'She', 'is',
'a', 'proven', 'draw', 'and', 'the', 'racists', 'on', 'the', 'upper', 'echelons', '.', 'just', 'ca', 'n't', 'handl
e', 'it', '.']
```

- Accuracy(model using split(" "))



- Loss(model using split(" "))



5. (1%) 請比較 RNN 與 BOW 兩種不同 model 對於 "Today is hot, but I am happy." 與 "I am happy, but today is hot." 這兩句話的分數 (model output)，並討論造成差異的原因。

RNN:

Today is hot, but I am happy. [0.08012492]

I am happy, but today is hot. [0.29768157]

BOW + DNN:

Today is hot, but I am happy. [0.03130154]

I am happy, but today is hot. [0.03146962]

RNN model 給 I am happy, but today is hot. 的分數較高(不過還是0.5以下所以不算惡意評論)。可見 RNN 有學到語意的部份，就是第一句的情緒是比較開心的。而 BOW 做出來的結果兩句幾乎一模一樣(不一樣的部份是因為第一句的 today 是大寫，還有句點跟最後的那兩個字黏在一起所以有三個字不一樣)。會造成這個結果是因為 BOW 是沒有考慮到語序的，所以應該也沒有辦法考慮字之間的關係，例如通常在句子中 "but" 後面才是重點，但是 BOW 根本不知道到底是哪些字在 but 後面所以當然做不到。

LSTM cell.

 $t=1$

$$g((0,1,0,3) \cdot (0,0,0,1) + 0) = 3$$

$$f((0,1,0,3) \cdot (100,100,0,0) - 10) = f(90) \approx 1$$

$$f((0,1,0,3) \cdot (-100,-100,0,0) + 110) = f(10) = 0.9999 \approx 1$$

$$C' = 3 \times 1 + 0 \times 1 = 3$$

$$y = f((0,1,0,3) \cdot (0,0,100,0) - 10) \times 3 = f(-10) \times 3 \approx 0.$$

 $t=2$

$$g((1,0,1,-2) \cdot (0,0,0,1) + 0) = -2$$

$$f((1,0,1,-2) \cdot (100,100,0,0) - 10) = f(90) = 1$$

$$f((1,0,1,-2) \cdot (-100,-100,0,0) + 110) = f(10) \approx 1$$

$$C' = -2 \times 1 + 3 \times 1 = 1$$

$$y = f((1,0,1,-2) \cdot (0,0,100,0) - 10) \cdot h(1) = 1$$

 $t=3$

$$g((1,1,1,4) \cdot (0,0,0,1) + 0) = 4$$

$$f((1,1,1,4) \cdot (100,100,0,0) - 10) = f(90) \approx 1$$

$$f((1,1,1,4) \cdot (-100,-100,0,0) + 110) = f(-90) \approx 0$$

$$C' = 4 \times 1 + (2) \times 0 = 4$$

$$y = f((1,1,1,4) \cdot (0,0,100,0) - 10) \cdot h(4) \approx 4$$

 $t=4$

$$g((0,1,1,0) \cdot (0,0,0,1) + 0) = 0$$

$$f((0,1,1,0) \cdot (100,100,0,0) - 10) = f(90) \approx 1$$

$$f((0,1,1,0) \cdot (-100,-100,0,0) + 110) = f(10) \approx 1$$

$$C' = 0 \times 1 + 4 \times 1 = 4$$

$$y = f((0,1,1,0) \cdot (0,0,100,0) - 10) \cdot h(4) = 4$$

$t=5$

$$g((0,1,1,0,1,2) \cdot (0,0,0,1) + 0) = 2$$

$$f((0,1,1,0,1,2) \cdot (100,100,0,0) - 10) = f(90) \cong 1$$

$$f((0,1,1,0,1,2) \cdot (-100,-100,0,0) + 110) = f(10) \cong 1$$

$$c' = 2 \times 1 + 4 \times 1 = 6$$

$$y = f((0,1,1,0,1,2) \cdot (0,0,100,0) - 10) \cdot 6 \cong 0$$

$t=6$

$$g((0,0,1,1,-4) \cdot (0,0,0,1) + 0) = -4$$

$$f((0,0,1,1,-4) \cdot (100,100,0,0) - 10) \cong 0$$

$$f((0,0,1,1,-4) \cdot (-100,-100,0,0) + 110) \cong 1$$

$$c' = -4 \times 0 + 6 \times 1 = 6.$$

$$y = f((0,0,1,1,-4) \cdot (0,0,100,0) - 10) \cdot 6 \cong 6.$$

$t=7$.

$$g((1,1,1,1,1) \cdot (0,10,0,1) + 0) = 1$$

$$f((1,1,1,1,1) \cdot (100,100,0,0) - 10) \cong 1$$

$$f((1,1,1,1,1) \cdot (-100,-100,0,0) + 110) \cong 0.$$

$$c' = 1 \times 1 + 6 \times 0 = 1$$

$$y = f((1,1,1,1,1) \cdot (0,0,100,0) - 10) \cdot 1 = 1$$

$t=8$

$$g((1,0,1,1,2) \cdot (0,10,0,1) + 0) = 2$$

$$f((1,0,1,1,2) \cdot (100,100,0,0) - 10) \cong 1$$

$$f((1,0,1,1,2) \cdot (-100,-100,0,0) + 110) \cong 1$$

$$c' = 2 \times 1 + 1 \times 1 = 3.$$

$$y = f((1,0,1,1,2) \cdot (0,0,100,0) - 10) \times 3 = 3.$$

Word Embedding

$$L = -\log \pi_{c \in C} \frac{\exp(u_c)}{\sum_{i \in V} \exp(u_i)}$$

$$\begin{aligned}\frac{\partial L}{\partial w_{ij}} &= \sum_{k=1}^V \sum_{c=1}^C \frac{\partial L}{\partial u_{ck}} \frac{\partial u_{ck}}{\partial w_{ij}} \\ &= \sum_{c=1}^C \frac{\partial L}{\partial u_{cj}} \frac{\partial u_{cj}}{\partial w_{ij}}\end{aligned}$$

$$= \sum_{c=1}^C (-\delta_{jj^*} + y_{cj}) \left(\sum_{k=1}^V w_{ki} x_k \right)$$

$$\frac{\partial L}{\partial u_{cj}} = -\delta_{jj^*} + y_{cj}, \text{ where } \delta_{jj^*}$$

$$= \begin{cases} 1, & \text{if } j = j^* \\ 0, & \text{otherwise} \end{cases}$$

$$\frac{\partial L}{\partial w_{ij}} = \sum_{k=1}^V \sum_{c=1}^C \frac{\partial L}{\partial u_{ck}} \frac{\partial}{\partial w_{ij}} \left(\sum_{m=1}^N \sum_{l=1}^V \bar{w}_{mk} w_{lm} x_l \right)$$

$$= \sum_{k=1}^V \sum_{c=1}^C (-\delta_{kj^*} + y_{ck}) w_{jk}^T x_i.$$