

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*

```
RNN(  
  (embedding): Embedding(5231, 100)  
  (gru): GRU(100, 256, batch_first=True, dropout=0.5, bidirectional=True)  
  (fc1): Sequential(  
    (0): Linear(in_features=512, out_features=1, bias=True)  
    (1): Sigmoid()  
  )  
)
```

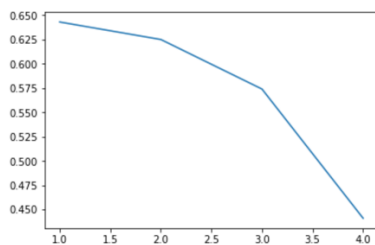
以上是我 RNN 模型的架構

先接一層 torch 的 gru 再接上 FCN

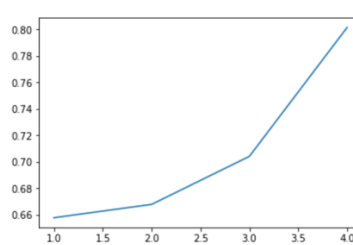
使用 Word2Vec 做出 word embedding

只 train 4 個 epoch

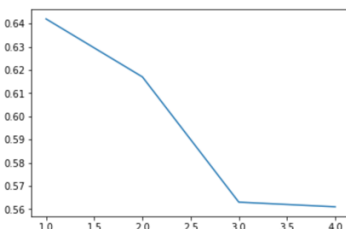
Train loss



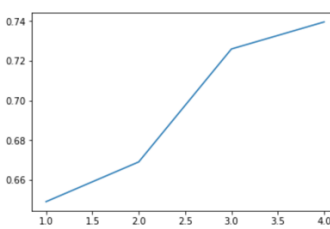
Train acc



Validation loss



Validation acc



最後在 kaggle 上 acc:

Public: 0.77209

Private: 0.77674

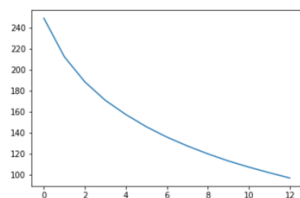
2. (1%) 請實作 BOW+DNN 模型，敘述你的模型架構，回報模型的正确率並繪出訓練曲線*。

```
BoWClassifier(  
    (linear): Linear(in_features=27639, out_features=2, bias=True)  
)
```

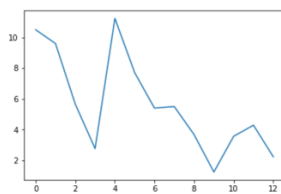
模型非常簡單就只有一層

Input 是 bow 的 size

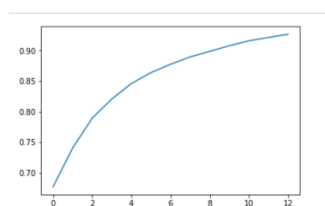
Train loss,



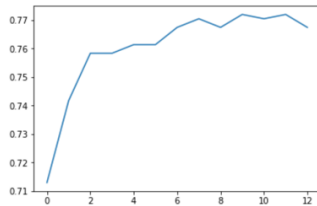
Validation loss



Train acc



Validation acc



最後在 kaggle 上 acc:

Public: 0.78139

Private: 0.80232

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。

我把比較沒用的字去掉像是

'@user', '@user', '..', '...', '@', '*', '#', '&', 'URL', ',', ' ', ' '

等等這些符號並不影響語義反而會造成干擾

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。

有做斷詞在 kaggle 上 private ,public 的正確率為

0.77674 與 0.77441

沒做斷詞只用空白分開的正確率為

0.75581 與 0.76744

可以看出做了斷詞可以讓結果比較好

可能是因為如果某些英文單字如果沒有適當的分開的話會造成 rnn 模型的誤解

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。

BOW :

使用 bow 做出來兩句話的分數相同為(dnn output size 為 2)

[-0.2153, -1.6416]

[-0.2153, -1.6416]

其中每個值為 log_probability 第一個為非惡意留言

所以這句話的 output label 為 0

因為 bag of word 是計算字出現的次數所以沒有先後關係兩串字對於 model 來說

是一模一樣的 input

RNN:

使用 RNN 做出來的分數分別為

[0.0042]. [0.0002]

大於 0.5 為惡意

可以看出兩句的確都不是惡意留言但是先後順序會造成 out put 分數不同

基本上都是很小的一個分數但是硬要說的或我在猜可能跟 happy 與 but 這個字出現的時機有

關, 第一句本來是開心的但後面接了個 but 可能會造成 rnn 認為這句或可能是往比

較不好的方向。

$$Z = wX + b \quad Z^A = w_A X + b_A \quad Z^I = w_I X + b_I \quad Z^O = w_O X + b_O$$

E	1	2	3	4	5	6	7	8
Z	3	-2	4	0	2	-4	1	2
Z^A	90	90	190	90	90	-10	190	90
Z^I	10	10	-90	10	10	110	-90	10
Z^O	-10	90	90	90	-10	90	90	90
$f(Z_i)$	1	1	1	1	1	0	1	1
$g(Z)$	3	-2	4	0	2	-4	1	2
c	0	3	1	4	4	6	6	1
$s(Z_i)$	1	1	0	1	1	1	0	1
$h(c)$	3	1	4	4	6	6	1	3
$f(Z_0)$	0	1	1	1	0	1	1	1
c'	3	1	4	4	6	6	1	3
g_i	0	1	4	4	0	6	1	3

$$y_i = 01440613$$

$$2. \quad \frac{\partial L}{\partial \theta_i} = -\log \frac{\exp(\theta_i)}{\sum_{j=1}^N \exp(\theta_j)} = -\frac{1}{\sum_{j=1}^N \exp(\theta_j)} \exp(\theta_i)$$

$$\frac{\partial L}{\partial \theta_i} = -\frac{\exp(\theta_i)}{\sum_{j=1}^N \exp(\theta_j)}$$

$$\frac{\partial L}{\partial \theta_i} = -\frac{\exp(\theta_i)}{\sum_{j=1}^N \exp(\theta_j)}$$

$$\frac{\partial L}{\partial \theta_i} = -\frac{\exp(\theta_i)}{\sum_{j=1}^N \exp(\theta_j)}$$

$$\frac{\partial L}{\partial \theta_i} = -\frac{\exp(\theta_i)}{\sum_{j=1}^N \exp(\theta_j)}$$

where δ_{ij} is Kronecker delta $\begin{cases} 1 & \text{if } i=j \\ 0 & \text{else} \end{cases}$

$$\frac{\partial L}{\partial w_{ij}} = \sum_{k=1}^N w_{ik} x_k$$

$$\Rightarrow \frac{\partial L}{\partial w_{ij}} = \sum_{k=1}^N (-\delta_{ik} + y_{ik}) \left(\sum_{l=1}^N w_{lj} x_l \right)$$

$$\Rightarrow \frac{\partial L}{\partial w_{ij}} = \sum_{k=1}^N (-\delta_{ik} + y_{ik}) \left(\sum_{l=1}^N w_{lj} x_l \right)$$

$$w_{ij} = \sum_{k=1}^N \sum_{l=1}^N w_{kl} w_{lj} x_k$$

$$\frac{\partial L}{\partial w_{ij}} = \frac{\partial L}{\partial w_{ij}} - w_{ij} x_j$$

$$\frac{\partial L}{\partial w_{ij}} = \sum_{k=1}^N \sum_{l=1}^N (-\delta_{ik} + y_{ik}) w_{lj} x_l$$