Homework 4 - Malicious Comments Identification

資工系博士班二年級 D06922023 顏志軒

2018 年 12 月 21 日

Problem 1. (0.5%) 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 *。(0.5%) 請實作 BOW+DNN 模型，敘述你的模型架構，回報正確率並繪出訓練曲線。

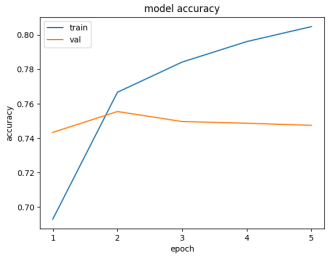
* 訓練曲線 (Training curve): 顯示訓練過程的 loss 或 accuracy 變化。橫軸為 step 或 epoch，縱軸為 loss 或 accuracy。

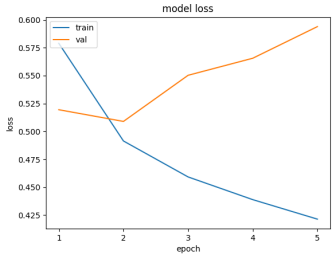
(a) 我實做的 RNN 架構為:

Layer (type)	Output Shape	Param #
embedding_1 (Embedding)	(None, 40, 192)	7560192
lstm_1 (LSTM)	(None, 256)	459776
dense_1 (Dense)	(None, 256)	65792
dropout_1 (Dropout)	(None, 256)	0
dense_2 (Dense)	(None, 1)	257

其中 embedding 層使用 word2vec 對訓練集與測試集的留言訓練出 192 維向量的權重作為參數，輸入字彙數目為 40。

訓練過程中的 accuracy 與 loss 分別為:



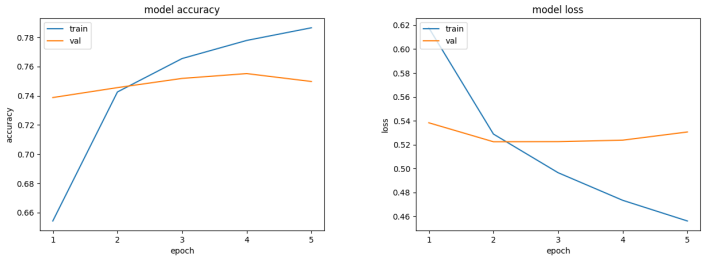


在 epoch 2 時 validation loss 最低，accuracy 為 0.7555。

(b) 我實做的 BOW+DNN 架構為:

Layer (type)	Output Shape	Param #
embedding_1 (Embedding)	(None, 40, 192)	7560192
dropout_1 (Dropout)	(None, 40, 192)	0
conv1d_1 (Conv1D)	(None, 36, 64)	61504
max_pooling1d_1 (MaxPooling1D)	(None, 12, 64)	0
flatten_1 (Flatten)	(None, 768)	0
dense_1 (Dense)	(None, 64)	49216
dropout_2 (Dropout)	(None, 64)	0
dense_2 (Dense)	(None, 1)	65

訓練過程中的 accuracy 與 loss 分別為:



在 epoch 2 時 validation loss 最低，accuracy 為 0.7456

兩種架構下，RNN 的準確率略高於 BOW，可能是因為 RNN 有考慮字彙前後的關聯。

Problem 2. (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。

我對深度學習模型做了以下調整:

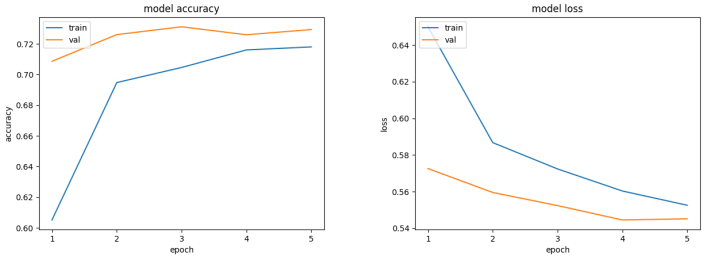
1. 調整 embedding 層輸入長度。經過多次嘗試，發現輸入長度為 40 左右的效果較佳，可能是測試資料中句子的平均 word 數約為 40。

2. 將使用 word2vec 的權重作為參數的 embedding 層的 trainable 改為 True。經過此調整準確率由約大於 50 提高到約大於 70，可能是因為 word2vec 僅考慮詞彙之間的關係，而沒有考慮詞彙與留言是否惡意的關聯。

3. 將 Conv+Dense 改為 LSTM+Dense，準確率提高約 0.01。有些微提昇可能是因為 LSTM 有考慮句子中前後詞彙的關係，而提昇的幅度不大的原因應該是惡意留言用抓關鍵字的方式也能夠有效的判斷。

Problem 3. (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。

不使用結巴斷詞，以 BOW+DNN 架構訓練所得的訓練過程如下：



在 epoch 4 時的 validation loss 最低，accuracy 為 0.7259，較有做斷詞的方法準確度低 2%。造成這種差異的原因，可能是因為不使用斷詞時，一個惡意詞會被分開考慮，因此不容易從原始留言中偵測出惡意的特徵。

Problem 4. (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。

兩句話在不同模型下的分數分別為：

句子	RNN	BOW
在說別人白痴之前，先想想自己	0.532	0.418
在說別人之前先想想自己，白痴	0.543	0.501

理論上 RNN 有考慮字彙之間的關係，應該較能分辨出這兩句話惡意程度的不同，但實驗結果卻完全相反，可能是因為訓練集中已經有類似的資料，

Problem 5. (1%) In this exercise, we will train a binary classifier with AdaBoost algorithm on the data shown in the table. Please use decision stump as the base classifier. Perform AdaBoost algorithm for T = 3 iterations. For each iteration (t = 1, 2, 3), write down the weights u_n^t used for training, the weighted error rate ϵ_t, scaling coefficient α_t, and the classification function $f_t(x)$. The initial weights u_n^1 are set to 1 (n = 0, 1, ..., 9). Please refer to the course slides for the definitions of the above notations. Finally, combine the three classifiers and write down the final classifier.

x	0	1	2	3	4	5	6	7	8	9
y	+	-	+	+	+	-	-	+	-	-

$T = 1$ 時，decision stump 的 20 種結果與加權錯誤率分別為：

```
 - - - - - - - - - - 0.5
+ - - - - - - - - - 0.4
+ + - - - - - - - - 0.5
+ + + - - - - - - - 0.4
+ + + + - - - - - - 0.3
+ + + + + - - - - - 0.2
+ + + + + + - - - - 0.3
+ + + + + + + - - - 0.4
+ + + + + + + + - - 0.3
+ + + + + + + + + - 0.4
+ + + + + + + + + + 0.5
- + + + + + + + + + 0.6
- - + + + + + + + + 0.5
- - - + + + + + + + 0.6
- - - - + + + + + + 0.7
- - - - - + + + + + 0.8
- - - - - - + + + + 0.7
- - - - - - - + + + 0.6
- - - - - - - - + + 0.7
- - - - - - - - - + 0.6
```

$x \leq 4$ 判斷為 $+$ 的加權錯誤率最低為 $0.2 \circ d = \sqrt{\frac{1-0.2}{0.2}} = 2$，$\alpha_t = ln(d) = 0.693 \circ$ 新的權重為：

0.5, 2, 0.5, 0.5, 0.5, 0.5, 0.5, 2, 0.5, 0.5

$T = 2$ 時，decision stump 的 20 種結果與加權錯誤率分別為：

```
 - - - - - - - - - - 0.5
+ - - - - - - - - - 0.4375
+ + - - - - - - - - 0.6875
+ + + - - - - - - - 0.625
+ + + + - - - - - - 0.5625
```

```
+ + + + + - - - - - 0.5
+ + + + + + - - - - 0.5625
+ + + + + + + - - - 0.625
+ + + + + + + + - - 0.375
+ + + + + + + + + - 0.4375
+ + + + + + + + + + 0.5
+ + + + + + + + + + 0.5
- + + + + + + + + + 0.5625
- - + + + + + + + + 0.3125
- - - + + + + + + + 0.375
- - - - + + + + + + 0.4375
- - - - - + + + + + 0.5
- - - - - - + + + + 0.4375
- - - - - - - + + + 0.375
- - - - - - - - + + 0.625
- - - - - - - - - + 0.5625
- - - - - - - - - - 0.5
```

$x \leq 1$ 判斷為-的加權錯誤率最低為 0.3125。$d = \sqrt{\frac{1-0.3125}{0.3125}} = 1.48$, $\alpha_t = ln(1.48) = 0.394$。新的權重為:

0.742 1.348 0.337 0.337 0.337 0.742 0.742 1.348 0.742 0.742

$T = 3$ 時, decision stump 的 20 種結果與加權錯誤率分別為:

```
 - - - - - - - - - - 0.418
+ - - - - - - - - - 0.318
+ + - - - - - - - - 0.500
+ + + - - - - - - - 0.455
+ + + + - - - - - - 0.409
+ + + + + - - - - - 0.364
+ + + + + + - - - - 0.464
+ + + + + + + - - - 0.564
+ + + + + + + + - - 0.382
+ + + + + + + + + - 0.482
+ + + + + + + + + + 0.582
```

```
+ + + + + + + + + + 0.582
- + + + + + + + + + 0.682
- - + + + + + + + + 0.500
- - - + + + + + + + 0.545
- - - - + + + + + + 0.591
- - - - - + + + + + 0.636
- - - - - - + + + + 0.536
- - - - - - - + + + 0.436
- - - - - - - - + + 0.618
- - - - - - - - - + 0.518
- - - - - - - - - - 0.418
```

$x \leq 0$ 判斷為 $+$ 的加權錯誤率最低為 $0.318 \circ d = \sqrt{\frac{1-0.318}{0.318}} = 1.464$, $\alpha_t = ln(d) = 0.381 \circ$

Problem 6. (1%) In this exercise, we will simulate the forward pass of a simple LSTM cell. Figure.1 shows a single LSTM cell, where z is the cell input, z_i, z_f, z_o are the control inputs of the gates, c is the cell memory, and f, g, h are activation functions. Given an input x, the cell input and the control inputs can be calculated by

$$z = w \cdot x + b$$
$$z_i = w_i \cdot x + b_i$$
$$z_f = w_f \cdot x + b_f$$
$$z_o = w_o \cdot x + b_o$$

Where w, w_i, w_f, w_o are weights and b, b_i, b_f, b_o are biases. The final output can be calculated by

$$y = f(z_o)h(c')$$

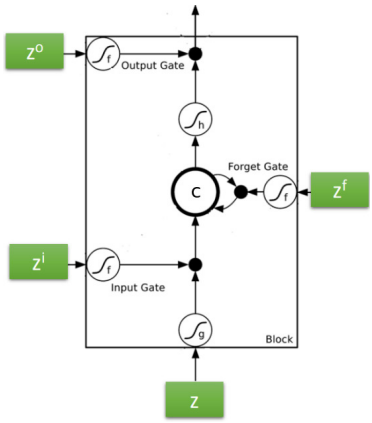
where the value stored in cell memory is updated by

$$c' = f(z_i)g(z) + cf(z_f)$$

Given an input sequence $x_t (t = 1, 2, \ldots, 8)$, please derive the output sequence y_t. The input sequence, the weights, and the activation functions

are provided below. The initial value in cell memory is 0. Please note that your calculation process is required to receive full credit.

$$w = [0, 0, 0, 1] \quad , b = 0$$
$$w_i = [100, 100, 0, 0] \quad , b_i = -10$$
$$w_f = [-100, -100, 0, 0] \quad , b_f = 110$$
$$w_o = [0, 0, 100, 0] \quad , b_o = -10$$



t	1	2	3	4	5	6	7	8
x_t	0	1	1	0	0	0	1	1
	1	0	1	1	1	0	1	0
	0	1	1	1	0	1	1	1
	3	-2	4	0	2	-4	1	2

$t = 0$ 時, $z = 3+0 = 3$, $g(z) = 0.953$, $z_i = 100-10 = 90$, $f(z_i) = 1.000$, $z_f = -100+110 = 10$, $f(z_f) = 1.000$, $c = 1.000 \times 0.953 + 0.000 \times 1.000 = 0.953$, $h(c) = 0.7216325609518421$, $z_o = 0 - 10 = -10$, $g(z_o) = 0.000$, $y = 0.000 \times 0.722 = 0.000$

$t = 1$ 時, $z = -2+0 = -2$, $g(z) = 0.119$, $z_i = 100-10 = 90$, $f(z_i) = 1.000$, $z_f = -100+110 = 10$, $f(z_f) = 1.000$, $c = 1.000 \times 0.119 + 0.953 \times 1.000 = 1.072$, $h(c) = 0.7449264975556411$, $z_o = 100 - 10 = 90$, $g(z_o) = 1.000$, $y = 1.000 \times 0.745 = 0.745$

$t = 2$ 時, $z = 4+0 = 4$, $g(z) = 0.982$, $z_i = 200-10 = 190$, $f(z_i) = 1.000$, $z_f = -200+110 = -90$, $f(z_f) = 0.000$, $c = 1.000 \times 0.982 + 1.072 \times 0.000 = 0.982$, $h(c) = 0.7275076135036415$, $z_o = 100 - 10 = 90$, $g(z_o) = 1.000$, $y = 1.000 \times 0.728 = 0.728$

$t = 3$ 時, $z = 0+0 = 0$, $g(z) = 0.500$, $z_i = 100-10 = 90$, $f(z_i) = 1.000$, $z_f = -100+110 = 10$, $f(z_f) = 1.000$, $c = 1.000 \times 0.500 + 0.982 \times 1.000 = 1.482$, $h(c) = 0.8148698338145558$, $z_o = 100 - 10 = 90$, $g(z_o) = 1.000$, $y = 1.000 \times 0.815 = 0.815$

$t = 4$ 時, $z = 2+0 = 2$, $g(z) = 0.881$, $z_i = 100-10 = 90$, $f(z_i) = 1.000$, $z_f = -100+110 = 10$, $f(z_f) = 1.000$, $c = 1.000 \times 0.881 + 1.482 \times 1.000 = 2.363$, $h(c) = 0.9139383334763549$, $z_o = 0 - 10 = -10$, $g(z_o) = 0.000$,

$y = 0.000 \times 0.914 = 0.000$
$t = 5$ 時, $z = -4 + 0 = -4$, $g(z) = 0.018$, $z_i = 0 - 10 = -10$, $f(z_i) = 0.000$, $z_f = 0 + 110 = 110$, $f(z_f) = 1.000$, $c = 0.000 \times 0.018 + 2.363 \times 1.000 = 2.363$, $h(c) = 0.9139383977009864$, $z_o = 100 - 10 = 90$, $g(z_o) = 1.000$, $y = 1.000 \times 0.914 = 0.914$
$t = 6$ 時, $z = 1 + 0 = 1$, $g(z) = 0.731$, $z_i = 200 - 10 = 190$, $f(z_i) = 1.000$, $z_f = -200 + 110 = -90$, $f(z_f) = 0.000$, $c = 1.000 \times 0.731 + 2.363 \times 0.000 = 0.731$, $h(c) = 0.6750375273768237$, $z_o = 100 - 10 = 90$, $g(z_o) = 1.000$, $y = 1.000 \times 0.675 = 0.675$
$t = 7$ 時, $z = 2 + 0 = 2$, $g(z) = 0.881$, $z_i = 100 - 10 = 90$, $f(z_i) = 1.000$, $z_f = -100 + 110 = 10$, $f(z_f) = 1.000$, $c = 1.000 \times 0.881 + 0.731 \times 1.000 = 1.612$, $h(c) = 0.8336642584279261$, $z_o = 100 - 10 = 90$, $g(z_o) = 1.000$, $y = 1.000 \times 0.834 = 0.834$

輸出序列為: [0.000, 0.745, 0.728, 0.815, 0.000, 0.914, 0.675, 0.834]