

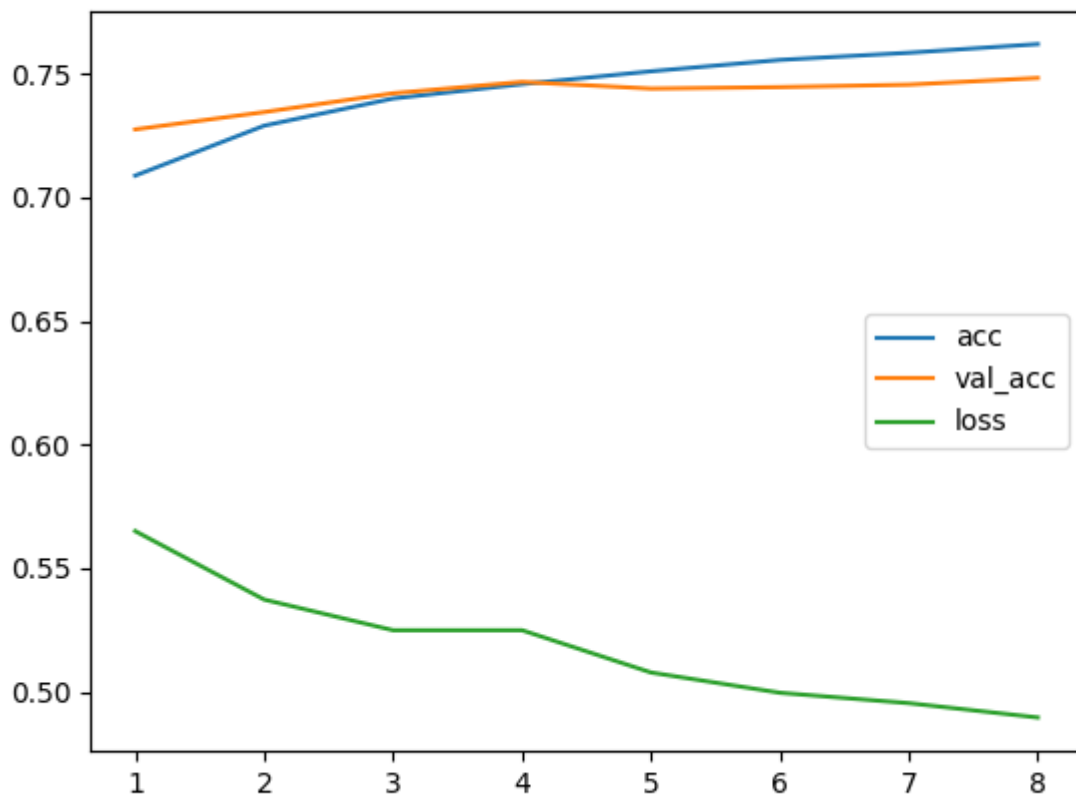
Homework 4 Report

Professor Pei-Yuan Wu EE5184 - Machine Learning b05902109 資工三 柯上優

1.

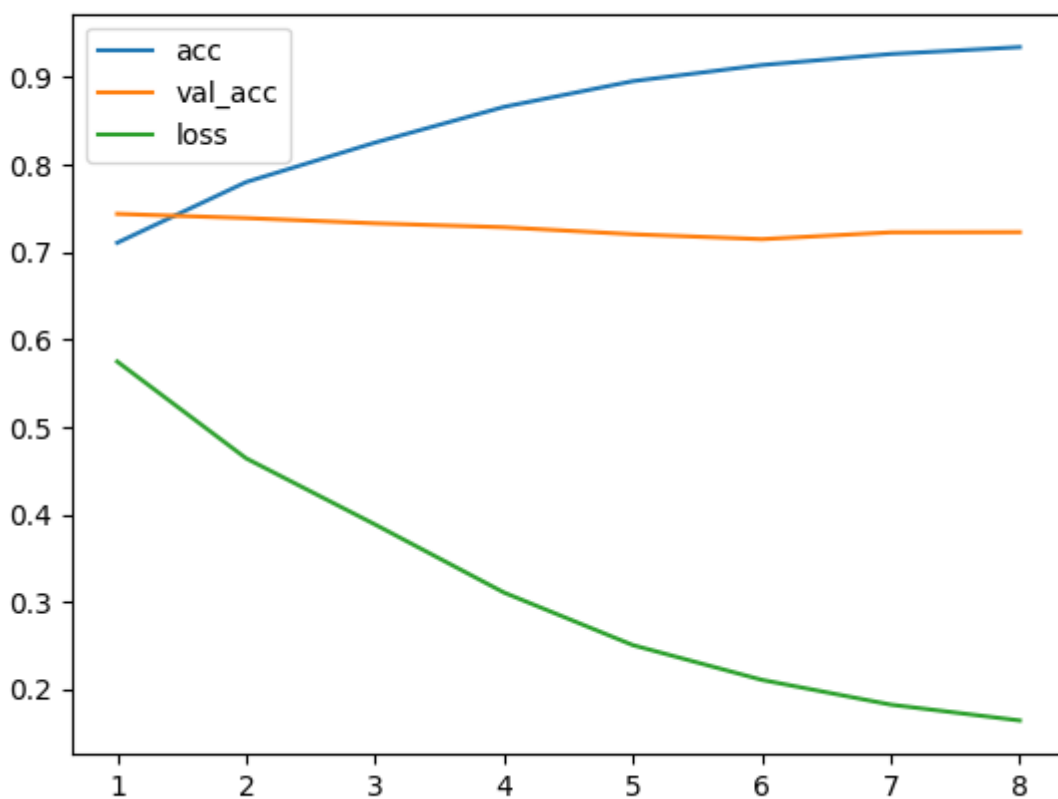
- data preprocessing
 - 使用jieba切割句子
 - 刪除Dcard的回應樓層(ex: b2、B3)
 - 使用trainX、testX的所有字訓練word2vec，dimension = 100
 - padding: 取句子長200，超過就切斷，不足則在前面補0

layer	parameter
lstm	256, return_sequences=True, dropout=0.3, recurrent_dropout=0.3
lstm	256, return_sequences=False, dropout=0.3, recurrent_dropout=0.3
Dense	256, activation=relu, dropout=0.5
Dense	1, Adam(lr=0.001, decay=1e-6, clipvalue=0.5)



- validation dataset是training data的前 $\frac{1}{10}$ 份
- data preprocessing
 - 同上。
 - BOW: 出現總次數大於10的，小於則捨去，共取了22760字詞。

layer	parameter
Dense	256, activation=relu, dropout=0.5
BatchNormalization	momentum=0.5
Dense	256, activation=relu, dropout=0.5
BatchNormalization	momentum=0.5
Dense	1, Adam(lr=0.001, decay=1e-6, clipvalue=0.5)



2.

- 最一開始我的Word2Vec的dimension只有32，做不太成功，val_acc = 0.5~0.6，後來經過同學建議而改成100，public_acc = 0.73。
 - 這顯示了dimension太小可能會降低字詞的辨識度，無法訓練。
- 接著我將原本的一層lstm增加到兩層lstm，public_acc = 0.74，此時仍尚未通過strong baseline。

- 適時的增加模型複雜度，可以提高準確率
- 後來我想起我的word2vec只有使用trainX中的詞，而改成將trainX、testX都拿來訓練，public_acc = 0.75
 - word2vec是unsupervised算法，若使用的字詞太少，OOV的狀況太多會遺失重要的feature。

3.

model	RNN+jieba	RNN+splitAllwords
valid_acc	0.7483	0.7356

- 觀察到沒有用jieba斷詞，準確度低於有斷詞
- 我認為，對於中文字，兩個相連的字，其意義多變且遠大於單獨一個文字，所以將中文做適時的斷詞有益於訓練與精準度。

4.

資料	RNN	BOW
"在說別人白痴之前，先想想自己"	0.4219915	0.9519825
"在說別人之前先想想自己，白痴"	0.6360231	0.9519825

- 我認為，關鍵字是"白癡"。
- 對於第一筆資料，RNN可能會隨著時間忘記，但是BOW發現這個字有出現後會馬上表現出來。
- 對於第二筆資料，RNN會清楚記得最後一個關鍵字，而認為帶有惡意。
- 此外，BOW對於兩筆資料有相同的數值，是出自於斷字的相同與BOW的性質。

5.

- 定義: 以下 $sign(0) = 1$
- $t = 1, f_1(x) = -1 \times sign(x - 5), \epsilon_1 = 0.2, \alpha_1 = 0.6931$
 - $u_1 = [1, 1, 1, 1, 1, 1, 1, 1, 1, 1]$
- $t = 2, f_2(x) = 1 \times sign(x - 2), \epsilon_2 = 0.3125, \alpha_2 = 0.3942$
 - $u_2 = [0.5, 2, 0.5, 0.5, 0.5, 0.5, 0.5, 2, 0.5, 0.5]$
- $t = 3, f_3(x) = -1 \times sign(x - 1), \epsilon_3 = 0.3182, \alpha_3 = 0.3810$
 - $u_3 = [0.7416, 1.3483, 0.3370, 0.3370, 0.3370, 0.7416, 0.7416, 1.3483, 0.7416, 0.7416]$
- final predict, error = 0.1

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

6.

t	$f(z_i)$	$g(z)$	$f(z_f)$	C	C'	$f(z_o)$	y
1	1	3	1	0	3	0	0
2	1	-2	1	3	1	1	1
3	1	4	0	1	4	1	4
4	1	0	1	4	4	1	4
5	1	2	1	4	6	0	0
6	0	-4	1	6	6	1	6
7	1	1	0	6	1	1	1
8	1	2	1	1	3	1	3