

Homework 3 Report - Image Sentiment Classification

學號：B06209027 系級：大氣二 姓名:李冠勳

1 (1%)

請說明你實作的 CNN model，其模型架構、訓練過程和準確率為何？

一開始我嘗試resnet和VGG的架構來設計我的模型，但resnet的設計結果略遜色於VGG架構，因此後來全部採用VGG來設計我的model，最後採用4種model的ensemble，單一model的設計大致如下：

Layer (type)	filters/units	activation function
Conv2D	64	relu/leaky relu
Conv2D	64	relu/leaky relu
MaxPooling2D		
Conv2D	128	relu/leaky relu
BatchNormalize		
MaxPooling2D		
Conv2D	256	relu/leaky relu
BatchNormalize		
MaxPooling2D		
Conv2D	512	relu/leaky relu
MaxPooling2D		
Dense	512	relu
Dense	512	relu
Dense	7	softmax

Conv2D 有加入batchnormalize

MaxPooling2D 有加入dropout

Dense 也有加入dropout

- model1用relu，model2用leaky relu，model3和4分別深度。(結果以model1最好)
- ensemble採用分數最高和次高的model1及分數最高的model2,3,4共五個model合併
- 預處理：以26000比data當test剩下的2709比data當validation data並用 ImageDataGenerator將圖片旋轉,平移等等。
- 訓練過程：model約用100個epoch達到0.6的準確率，往後緩慢上升到0.68左右
- 單一model成績最好為

model	validation score	public score	private score
single model	0.70949	0.69852	0.72053

- 整體成績最好為

model	validation score	public score	private score
ensemble model	0.72794	0.71106	0.73585

2 (1%)

承上題，請用與上述 CNN 接近的參數量，實做簡單的 DNN model，其模型架構、訓練過程和準確率為何？試與上題結果做比較，並說明你觀察到了什麼？

- **DNN model**

Layer (type)	Output Shape	Param #
Flatten	2304	0
Dense	2048	4720640
Dense	512	1049088
Dense	128	65664
Dense	7	903
Activation	7	0

activation funcion : leaky relu
每一層都有用batch normalize
Total params: 5847047
Trainable params: 5841671
Non-trainable params: 5376

• CNN model

Layer (type)	Output Shape	Param #
Conv2D	(48, 48, 64)	1664
MaxPooling2D	(24, 24, 64)	0
Conv2D	(24, 24, 128)	73856
MaxPooling2D	(12, 12, 128)	0
Conv2D	(12, 12, 512)	590336
MaxPooling2D	(6, 6, 512)	0
Conv2D	(6, 6, 512)	2359808
MaxPooling2D	(3, 3, 512)	0
Dense	512	2359808
Dense	512	262656
Dense	7	2048

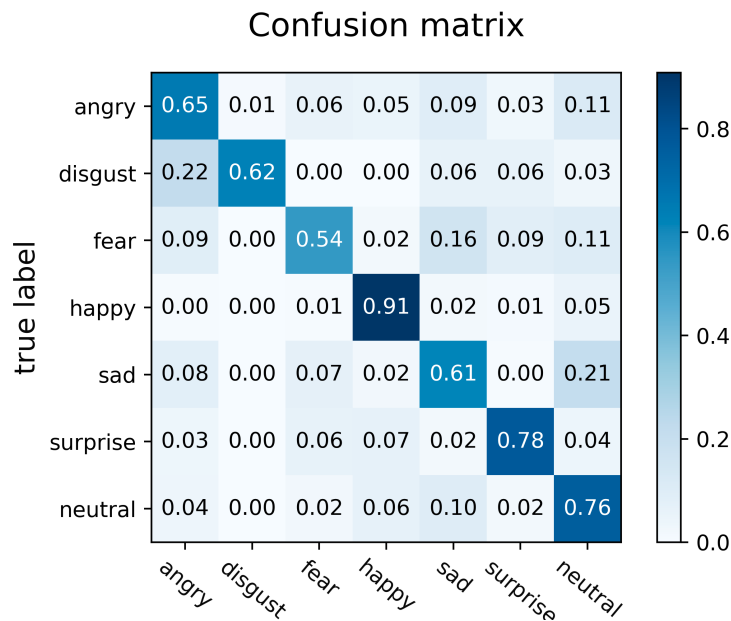
activation funcion : leaky relu
非maxpooling層都有用batch normalize
非conv2d層都有用batch normalize
Total params: 5,660,679
Trainable params: 5,656,199
Non-trainable params: 4,480
分別取訓練過程中最好的結果來評量

model	training score	validation score	public score	private score
DNN	0.4238	0.45995	0.13402	0.13290
CNN	0.7550	0.70469	0.68152	0.70465

- 可以發線DNN model在訓練時就已經不能得到很好的結果了，但沒想到我選validation成績最高的結果去test卻得到public和private比random猜還差的結果。
- 反觀CNN model參數雖差不多但得到的結果好許多
- 從上表的參數可以發現Dense層會增加許多的參數Conv2D層則少許多，因此可以用較少的參數量得更深
- 在訓練過程中可以發現DNN accuracy上升速度緩慢，到0.4左右就幾乎卡住了，而CNN則一直上升到0.6才比較緩慢

3 (1%)

觀察答錯的圖片中，哪些 class 彼此間容易用混？並說明你觀察到了什麼？
[繪出 confusion matrix 分析]



- 使用2709比validation data繪製上圖
- 從上圖可以發現disgust, fear, sad有比較低的準確率
- disgust比較容易分成angry, sad比較容易分成neutral可能是兩者表情起伏的位置與起伏方向相似
- happy的表情較誇張因此比較不容易分錯

4 (1.5%)

a. How many parameters are there in each layer(Hint: you may consider whether the number of parameter is related with)

- LayerA : $2 \times 2 \times 6 \times 5 + 6 = 126$
- LayerB : $2 \times 2 \times 6 \times 4 + 6 = 106$

b. How many multiplications/additions are needed for a forward pass(each layer).

- LayerA
 - multiplication : $6 \times (2 \times 2 \times 5) \times (\frac{8-2}{3} + 1)^2 = 1080$
 - addition : $6 \times (2 \times 2 \times 5 - 1) \times (\frac{8-2}{3} + 1)^2 = 1026$
- LayerB
 - multiplication : $4 \times (2 \times 2 \times 6) \times (\frac{2-2}{2} + 1)^2 = 96$
 - addition : $4 \times (2 \times 2 \times 6 - 1) \times (\frac{2-2}{2} + 1)^2 = 92$

c. What is the time complexity of convolutional neural networks?

假設從第1層到第m層，其中第i層的數據：

kernel size : (k_i, k_i)

channel size : c_i

input shape of each layer : (n_i, n_i)

padding : p_i

strides : (s_i, s_i)

$$\text{時間複雜度} : \sum_{i=1}^m c_i \left(\left\lfloor \frac{(n_i - k_i) + p_i}{s_i} \right\rfloor \times k_i^2 \right)^2 + 1$$

因為 k_i 與 p_i 很接近故消除，因此時間複雜度為 $O(\frac{cn^2k^4}{s^2})$

5 (1.5%)

使用numpy計算eigenvector，並使用pca計算principal components

a. 三個principal axes(取小數點兩位)：

- vector 1 = <0.62,0.59,0.52>
- vector 2 = <0.40, 0.34, -0.85>
- vector 3 = <0.67, -0.73, 0.03>

b. principal components :

- project to vector 1, vector 2 and vector 3

x	y	z		x	y	z		x	y	z
2.07	1.98	1.76		-0.59	-0.50	1.26		-0.48	-0.52	-0.02
6.04	5.76	5.12		0.02	0.01	-0.03		2.05	2.22	-0.08
8.40	8.02	7.12		-0.97	-0.82	2.06		-4.43	4.80	-0.18
4.90	4.68	4.15		-0.46	-0.39	0.99		-3.43	3.72	-0.14
7.63	7.28	6.47		2.01	1.70	-4.28		-4.64	5.02	-0.19
4.44	4.24	3.76		1.32	1.11	-2.81		1.25	-1.35	0.05
9.23	8.81	7.82		-0.55	-0.46	1.17		0.32	-0.35	0.01
4.37	4.17	3.70		1.22	1.03	-2.60		-2.59	2.80	-0.10
7.93	7.57	6.72		0.39	0.33	-0.83		2.67	-2.90	0.11
10.05	9.60	8.52		0.70	0.59	-1.49		-0.75	0.81	-0.03

c. 選兩個eigenvector, vector1和vector3

- Reconstruction

x	y	z
1.59	2.50	1.74
3.98	7.99	5.03
3.97	12.82	6.94
1.46	8.39	4.01
2.99	12.30	6.28
5.68	2.89	3.81
9.55	8.46	7.83
1.78	6.97	3.60
10.61	4.67	6.83
9.30	10.41	8.49

- L2-norm = 20.56