

Convolution Neural Networks

Date

2018.04.25

TA

Yen

Mail

Mail : a94763075@163.com

kernel = filter = feature

- `Conv2D(filters=9,kernel_size=(3,3), strides=1,padding='same')`

- $$\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}$$

1 _{x1}	1 _{x0}	1 _{x1}	0	0
0 _{x0}	1 _{x1}	1 _{x0}	1	0
0 _{x1}	0 _{x0}	1 _{x1}	1	1
0	0	1	1	0
0	1	1	0	0

Image

4		

Convolved
Feature

padding

- `Conv2D(filters=9, kernel_size=(3,3), strides=1, padding='same')`

with zero padding:

0 ₂	0 ₀	0 ₁	0	0	0	0
0 ₁	2 ₀	2 ₀	3	3	3	0
0 ₀	0 ₁	1 ₁	3	0	3	0
0	2	3	0	1	3	0
0	3	3	2	1	2	0
0	3	3	0	2	3	0
0	0	0	0	0	0	0

1	6	5
7	10	9
7	10	8

padding

- `Conv2D(filters=9, kernel_size=(3,3), strides=1, padding='valid')`

without padding:

3 ₀	3 ₁	2 ₂	1	0
0 ₂	0 ₂	1 ₀	3	1
3 ₀	1 ₁	2 ₂	2	3
2	0	0	2	2
2	0	0	0	1

12	12	17
10	17	19
9	6	14

Pooling layer

- `MaxPooling2D((2, 2),padding='same')`

Feature Map

6	4	8	5
5	4	5	8
3	6	7	7
7	9	7	2

Max-Pooling

Define loss function_L1_norm

```
from keras import backend as K
```

- `def loss(y_true,y_pred):`
- `return (K.mean(K.abs(y_true-y_pred)))`

$$S = \frac{1}{n} * \sum_{\{i=1\}}^n |y_i - f(x_i)|$$

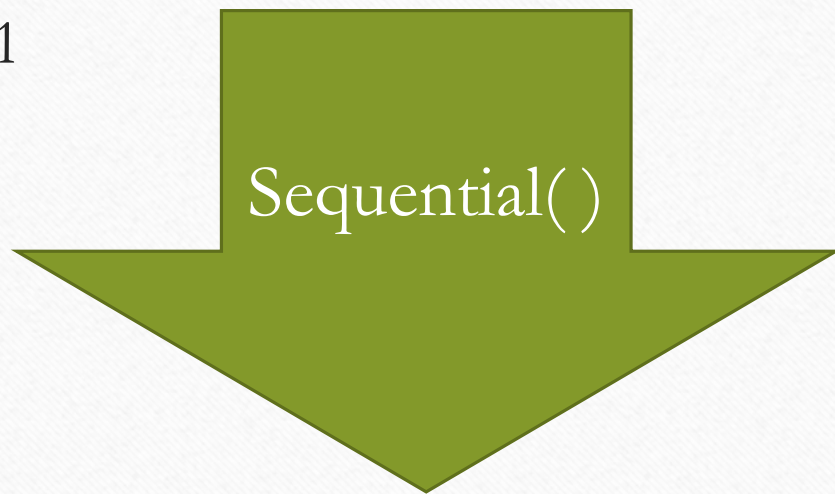
Mean Square Error(MSE)

$$S = \frac{1}{n} * \sum_{\{i=1\}}^n |y_i - f(x_i)|^2$$

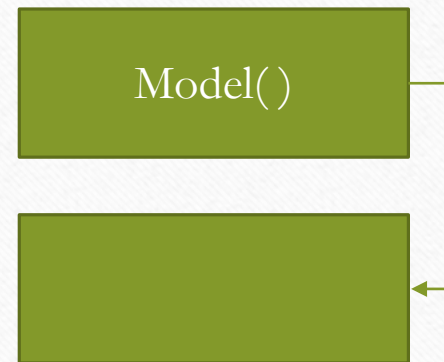
Model

- 2 ways to build model

• 1



2




```
image_width=256
image_height=256
input_channel=3

model = Sequential()
model.add(Conv2D(
    filters=9,
    kernel_size=(3,3),
    padding='same',
    input_shape=(image_width,image_height,input_channel),
    activation='relu'
))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Flatten())
model.add(Dense(10))
model.add(Activation('softmax'))

model.summary()
```

```
image_width=256
image_height=256
input_channel=3

input_shape = (image_width, image_height, input_channel)

inputs = Input(shape=input_shape)
Layer1 = Conv2D(9, (3, 3), strides=1, padding='same', activation='relu')(inputs)
Layer2 = MaxPooling2D(pool_size=(2,2))(Layer1)
Layer3 = Flatten()(Layer2)
Layer4 = Dense(10)(Layer3)
Layer5 = Activation('softmax')(Layer4)
model = Model(inputs=inputs, outputs=Layer5)

model.summary()
```


model.summary()

```
Layer (type)                 Output Shape              Param #
=====
conv2d_1 (Conv2D)            (None, 256, 256, 9)      252
max_pooling2d_1 (MaxPooling2 (None, 128, 128, 9)      0
flatten_1 (Flatten)          (None, 147456)           0
dense_1 (Dense)               (None, 10)                1474570
activation_1 (Activation)     (None, 10)                0
=====
Total params: 1,474,822
Trainable params: 1,474,822
Non-trainable params: 0
```

HW4: Predict “Push” and “Down”

Scoring 15%

- Code: must
- Report: 7%
- Kaggle: 8%

HW4: Predict “Push” and “Down”

Kaggle:

- Kaggle: <https://goo.gl/DQNxN4>
- **Deadline: 5/13 23:59**
- Upload limit 5 times/day
- Must over baseline
- Final Score = $3 + 5 * \frac{\text{Baseline} - \text{YourScore}}{\text{Baseline} - \text{MinScore}}$

HW4: Predict “Push”

Data

- :PTT(training:1800*20,testing:4000)



推 5 噓 0

HW4: Predict “Push” and “Down”

Setting About Baseline:

HW3 Embedding

+Conv1D

+Flatten

+Dense

+Activation

```
embedding_1 (Embedding)
conv1d_1 (Conv1D)
flatten_1 (Flatten)
dense_1 (Dense)
activation_1 (Activation)
```

HW4: Predict “Push” and “Down”

Submission Format

- id, good, bad

```
id,good,bad
0,41,0
1,38,3
2,23,2
3,37,0
4,15,2
5,43,5
6,16,3
```


HW4: Predict “Push” and “Down”

Loss LIMIT

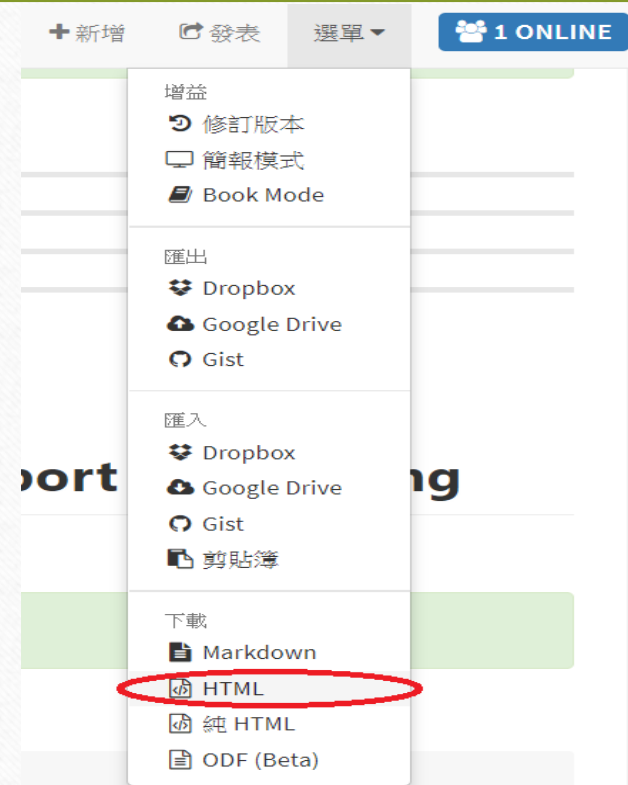
- You should define loss **by yourself**:
- α, β, γ you can free to adjust

$$loss = \frac{1}{n} \sum_{i=0}^n (\alpha |y_i - f(x_i)| + \beta |y_i - f(x_i)|^2 + \gamma |y_i - f(x_i)|^3)$$

HW4: Predict “Push” and “Down”

Report: 7%

- Don't submit word
- You have to submit .html
- Here is the report's example
- <https://hackmd.io/s/Byn-DMnnz>
- Word, No TypeSetting Report will get 0 %



HW4: Predict “Push” and “Down”

Report: 7%

- Model description (Draw and Explain and Loss)(3%)
- Preprocessing Explain(3%)
- How do you improve your performance (1%)

HW4: Predict “Push” and “Down”

Final Submission should be like as follow:



B10306478_陳XX.html



B10306478_陳XX.py

Q&A?