# DenseNet-Keras-novel.py

最基础的denseNet, k=12

|  |  |  |
| --- | --- | --- |
| Layers | Output Size | DenseNet-novel |
| Input | 32 x 32 x 3 |  |
| Convolution | 32 x 32 x 2k | 3x3, padding=same |
| Pooling | 16 x 16 x 2k | 2x2 ,MaxPooliing |
| Dense Block-B  (1) | 16 x 16 |  |
| Transition Layer  (1) | 8 x 8 | 1x1 conv , channel=k  2x2 MaxPooling |
| Dense Block-B  (2) | 8 x 8 |  |
| Transition Layer  (2) | 4 x 4 | 1x1 conv , channel=k  2x2 MaxPooling |
| Dense Block-B  (3) | 4 x 4 |  |
| Classification Layer | 1 x 1 | 4 x4 AveragePooling2D  10-d Full-connection,  softmax |

**训练细节：**

Batch\_size = 32

Epoch = 300

Data\_argumentation = False

Learning\_rate =0.01

Decay=1e-6

Optimization = RMSprop

Loss = categorical\_crossentropy

Metrics=[‘accuracy’]

**训练结果：（严重过拟合了）**

Train\_loss= 0.034， train\_acc= 0.992

Val\_loss= 1.9 ， val\_acc= 0.78

Test\_time= 2s/10,000 次

**增加L2正则：**

**Lamda\_reg = 0.001**