AlexNet应用于cifar数据集

1. 修改的AlexNet结构

1）AlexNet详细结构

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Modified AlexNet for Cifar(1FC) | | | | | |
| input | 32\*32\*3 | | | | |
| layer1 | Conv2d | kernel | channel | padding | stride |
| 11\*11\*3 | 64 | 5 | 4 |
| Relu | inplace=True | | | |
| MaxPool2d | kernel\_size | | stride | |
| 2\*2 | | 2 | |
| layer2 | Conv2d | kernel | channel | padding | stride |
| 5\*5\*64 | 192 | 2 | default |
| Relu | inplace=True | | | |
| MaxPool2d | kernel\_size | | stride | |
| 2\*2 | | 2 | |
| layer3 | Conv2d | kernel | channel | padding | stride |
| 3\*3\*192 | 384 | 1 | default |
| Relu | inplace=True | | | |
| layer4 | Conv2d | kernel | channel | padding | stride |
| 3\*3\*384 | 256 | 1 | default |
| Relu | inplace=True | | | |
| layer5 | Conv2d | kernel | channel | padding | stride |
| 3\*3\*256 | 256 | 1 | default |
| Relu | inplace=True | | | |
| MaxPool2d | kernel\_size | | stride | |
| 2\*2 | | 2 | |
| fully-connected | 256 -> 10 | | | | |

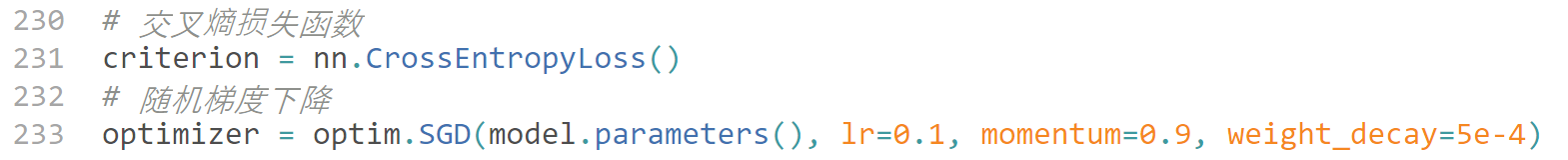
2）参数初始化

learning\_rate = 0.1

momentum = 0.9

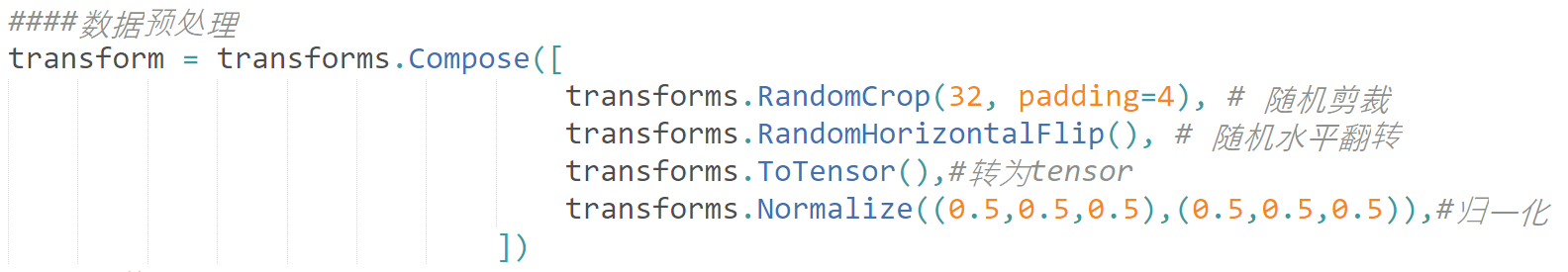
weight\_decay = 0.0005

损失函数使用交叉熵，训练过程使用带动量的随机梯度下降法。

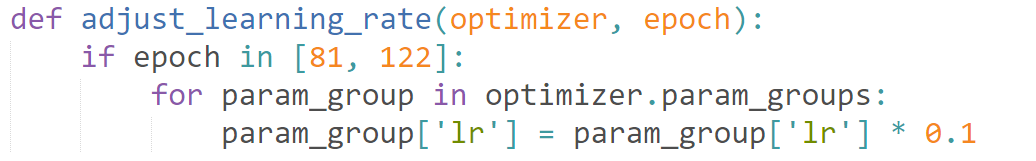


3）测试及网络参数调整

（1）增强数据集



（2）修改学习率



4）结果对比展示

（1）无数据增强，无学习率调整

|  |  |  |
| --- | --- | --- |
| Best Accuracy | | |
| train | Top-1 |  |
| Top-5 |  |
| test | Top-1 |  |
| Top-5 |  |

（2）无数据增强，有学习率调整

|  |  |  |
| --- | --- | --- |
| Best Accuracy | | |
| train | Top-1 |  |
| Top-5 |  |
| test | Top-1 |  |
| Top-5 |  |

（3）有数据增强，无学习率调整

|  |  |  |
| --- | --- | --- |
| Best Accuracy | | |
| train | Top-1 |  |
| Top-5 |  |
| test | Top-1 |  |
| Top-5 |  |

（4）有数据增强，有学习率调整

|  |  |  |
| --- | --- | --- |
| Best Accuracy | | |
| train | Top-1 |  |
| Top-5 |  |
| test | Top-1 |  |
| Top-5 |  |

2. 全连接层结构调整

1）AlexNet详细结构

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Modified AlexNet for Cifar(3FC) | | | | | |
| input | 32\*32\*3 | | | | |
| layer1 | Conv2d | kernel | channel | padding | stride |
| 11\*11\*3 | 64 | 5 | 4 |
| Relu | inplace=True | | | |
| MaxPool2d | kernel\_size | | stride | |
| 2\*2 | | 2 | |
| layer2 | Conv2d | kernel | channel | padding | stride |
| 5\*5\*64 | 192 | 2 | default |
| Relu | inplace=True | | | |
| MaxPool2d | kernel\_size | | stride | |
| 2\*2 | | 2 | |
| layer3 | Conv2d | kernel | channel | padding | stride |
| 3\*3\*192 | 384 | 1 | default |
| Relu | inplace=True | | | |
| layer4 | Conv2d | kernel | channel | padding | stride |
| 3\*3\*384 | 256 | 1 | default |
| Relu | inplace=True | | | |
| layer5 | Conv2d | kernel | channel | padding | stride |
| 3\*3\*256 | 256 | 1 | default |
| Relu | inplace=True | | | |
| MaxPool2d | kernel\_size | | stride | |
| 2\*2 | | 2 | |
| fully-connected | Dropout | | | | |
| Linear | 256->4096 | | | |
| Relu | inplace=True | | | |
| Dropout | | | | |
| Linear | 4096->4096 | | | |
| Relu | inplace=True | | | |
| Linear | 4096->10 | | | |

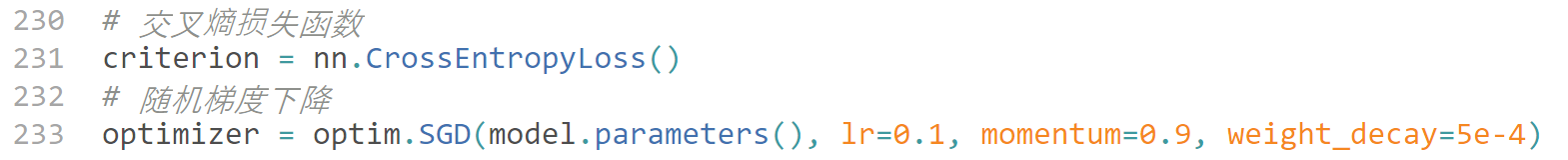
2）参数初始化

learning\_rate = 0.1

momentum = 0.9

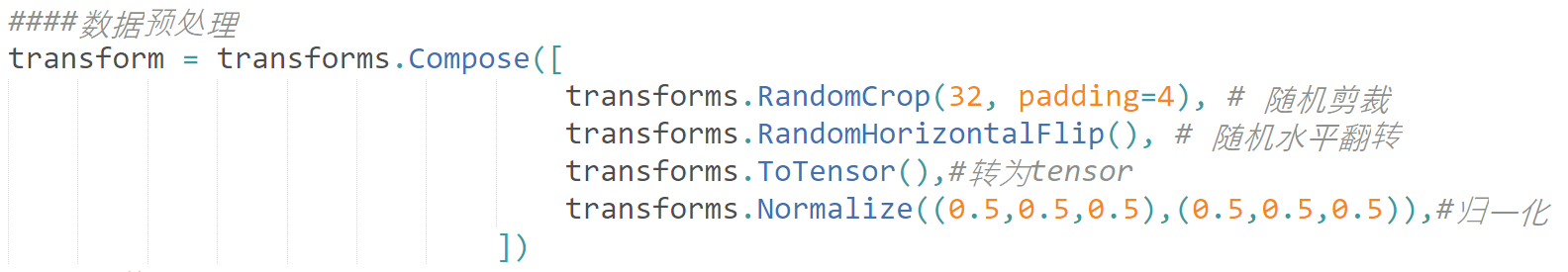
weight\_decay = 0.0005

损失函数使用交叉熵，训练过程使用带动量的随机梯度下降法。

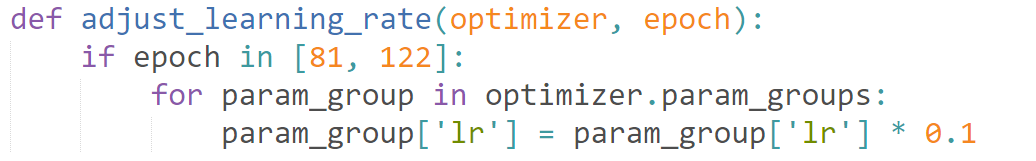


3）测试及网络参数调整

（1）增强数据集



（2）修改学习率



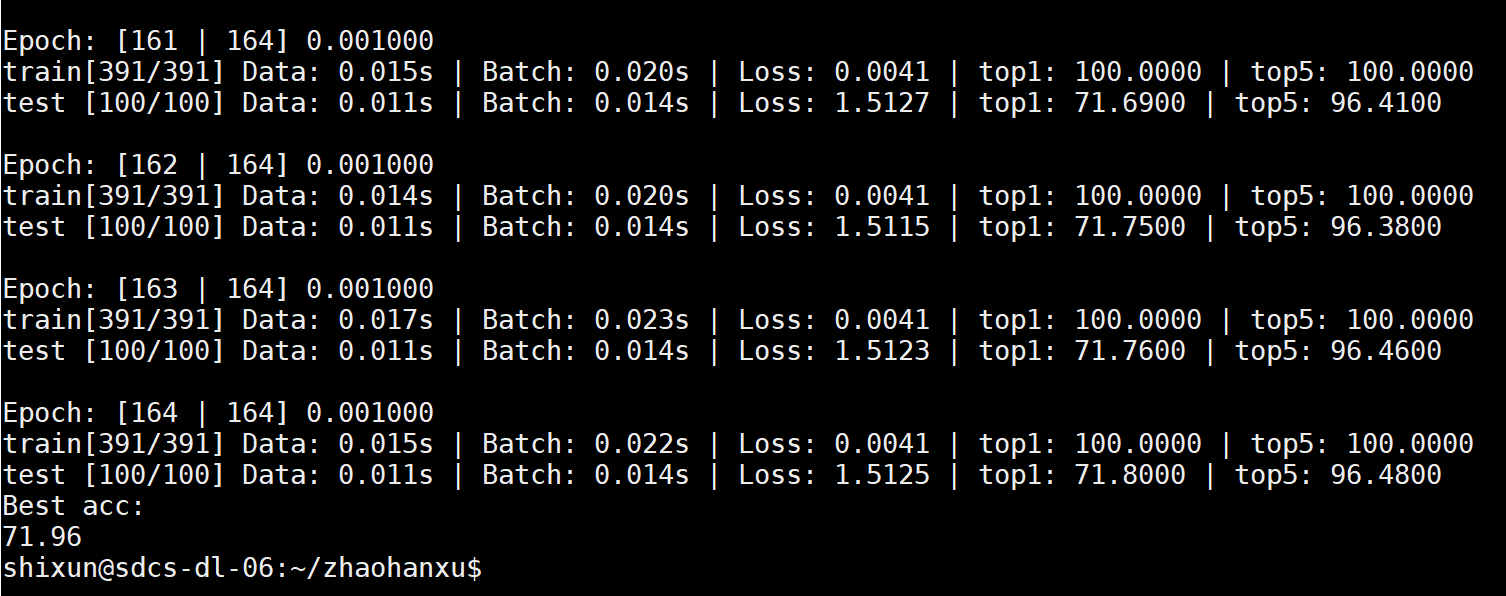
4）结果对比展示

（1）无数据增强，无学习率调整

|  |  |  |
| --- | --- | --- |
| Best Accuracy | | |
| train | Top-1 |  |
| Top-5 |  |
| test | Top-1 |  |
| Top-5 |  |

（2）无数据增强，有学习率调整

|  |  |  |
| --- | --- | --- |
| Best Accuracy | | |
| train | Top-1 |  |
| Top-5 |  |
| test | Top-1 |  |
| Top-5 |  |



（3）有数据增强，无学习率调整

|  |  |  |
| --- | --- | --- |
| Best Accuracy | | |
| train | Top-1 |  |
| Top-5 |  |
| test | Top-1 |  |
| Top-5 |  |

（4）有数据增强，有学习率调整

|  |  |  |
| --- | --- | --- |
| Best Accuracy | | |
| train | Top-1 |  |
| Top-5 |  |
| test | Top-1 |  |
| Top-5 |  |

