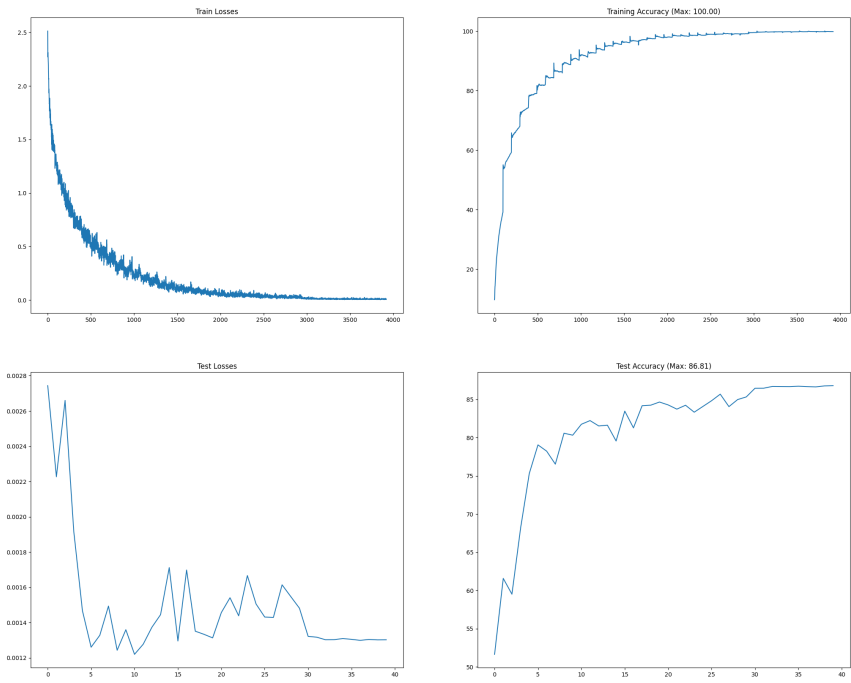


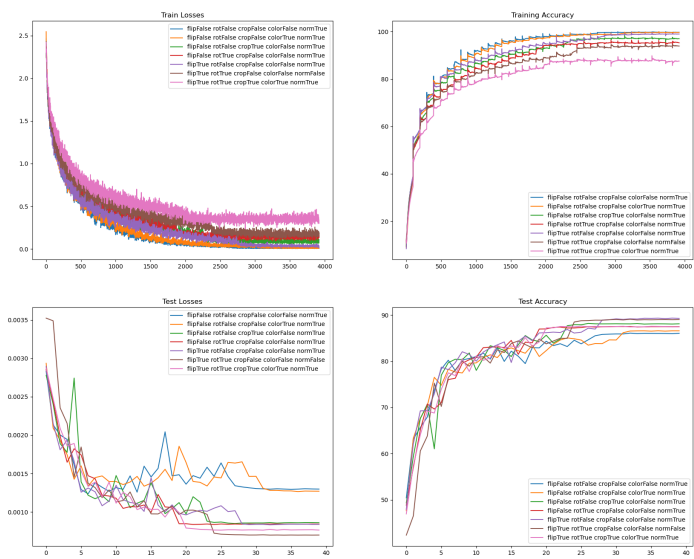
报告-PJ2-21302010040-叶天逸

任务一： RESNET



任务二： 数据增强

采用多种图像增强组合来测试:



1. 数据增强后，Test Loss 和准确率都更加稳定

2. EPOCH在30+基本就稳定了，以后可以取32来节约时间，如果发现没有收敛再增加
3. 我一开始用的是默认的resnet_SGD_ReduceLROnPlateau, 发现如下: 翻转效果最好

Test Accuracy: 89.28, File Path:

metrics_flipTrue_rotFalse_cropFalse_colorFalse_normTrue_resnet_SGD_ReduceLROnPlateau.npz

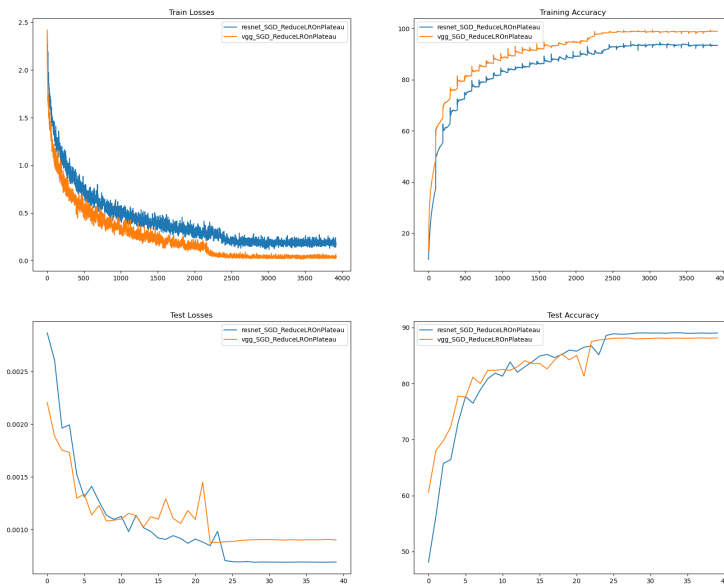
Test Accuracy: 89.00, File Path:

metrics_flipTrue_rotTrue_cropFalse_colorFalse_normFalse_resnet_SGD_ReduceLROnPlateau.npz

任务四：VGG

对比一下VGG和Resnet

感觉VGG激进一些, 最后结果还是resnet略高, VGG有点过拟合了。后面还有分析, vgg可以很不错



任务三：优化器和调度器, 加上网络

通过控制变量，绘制：这个图有点糊，我是利用分开的绘图分析的

1. 从最终效果上看, ReduceLROnPlateau 最好, vgg比较好
2. 冠军是: vgg_RMSprop_ReduceLROnPlateau
3. 列出最好的6名: (按照最后三次准确率的均值)

Test Accuracy: 89.58, File Path: metrics_vgg_RMSprop_ReduceLROnPlateau.npz

Test Accuracy: 89.02, File Path: metrics_resnet_SGD_ReduceLROnPlateau.npz

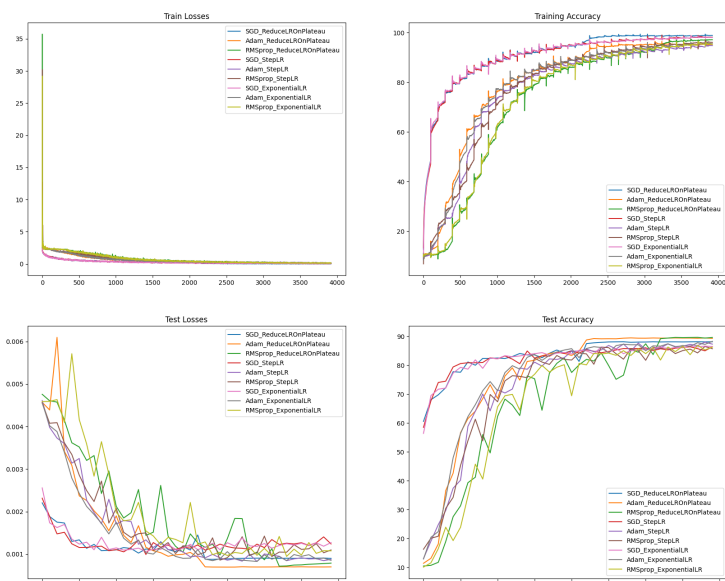
Test Accuracy: 88.09, File Path: metrics_vgg_SGD_ReduceLROnPlateau.npz

Test Accuracy: 87.78, File Path: metrics_vgg_Adam_ExponentialLR.npz

Test Accuracy: 87.67, File Path: metrics_vgg_Adam_ReduceLROnPlateau.npz

Test Accuracy: 87.55, File Path: metrics_resnet_SGD_StepLR.npz

全部



附加: 最优

1. 翻转可以带来优质, 结合vgg, RMSprop, 平台减小, 我们可以得到最优解

Test Accuracy: 89.31, File Path:

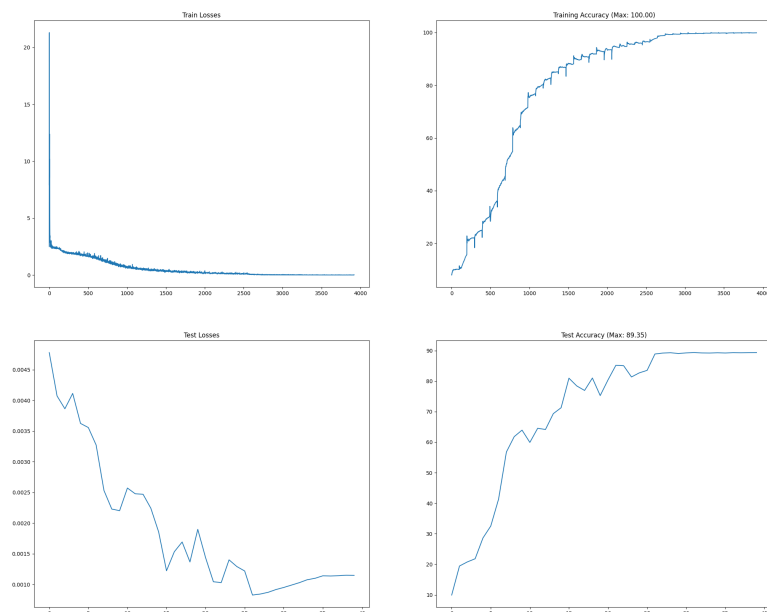
metrics_flipTrue_rotFalse_cropFalse_colorFalse_normTrue_vgg_RMSprop_ReduceLROnPlateau.npz

Test Accuracy: 89.28, File Path:

metrics_flipTrue_rotFalse_cropFalse_colorFalse_normTrue_resnet_SGD_ReduceLROnPlateau.npz

我这里的冠军产生, 最高89.31,

flipTrue_rotFalse_cropFalse_colorFalse_normTrue_vgg_RMSprop_ReduceLROnPlateau



技术

1. 持久化: 通过保存中间结果可以从非0 epoch开始迭代, 防止 "中断", "未收敛时需要加长epoch" 等情况

```
=====
Total params: 14,728,266
Trainable params: 14,728,266
Non-trainable params: 0
-----
Input size (MB): 0.01
Forward/backward pass size (MB): 6.57
Params size (MB): 56.18
Estimated Total Size (MB): 62.76
-----
EPOCHS : 32
Loss=0.10425398498773575 Batch_id=97 Accuracy=97.67: 100%|██████████| 98/98 [00:04<00:00, 24.23it/s]

Test set: Average loss: 0.0014, Accuracy: 8501/10000 (85.01%)

EPOCHS : 33
Loss=0.11045122146606445 Batch_id=97 Accuracy=96.76: 100%|██████████| 98/98 [00:03<00:00, 24.80it/s]

Test set: Average loss: 0.0012, Accuracy: 8601/10000 (86.01%)

EPOCHS : 34
Loss=0.06593284755945206 Batch_id=97 Accuracy=96.96: 100%|██████████| 98/98 [00:03<00:00, 24.51it/s]
```

2. 持久化: 通过保存metric然后作画可以避免因为作画程序问题导致功亏一篑
3. 通过参数包裹的程序来实现大批量的测试

感想:

1. 提交了notebook文件, 有关更多实现和代码请查看其中.
2. 进入到深度学习阶段, 模型训练的算力开始变得关键, 对于一个epoch, colab需要30s, moda的PAI-DSW需要13s, 而我在autoDL上花钱租只要4s, 也就是对于18个任务, 只需要48分钟. 花钱会有一些心疼, 我在想, 也许样本量是可以减少的, 但是为了更好的结果, 就是要数据, 于是算力, 于是钱。。花钱爽!
3. 但是网上的不一定能保存模型, 因此还是很浪费时间, 因为网络的问题很多坑。
4. 缺陷: 尚未检测不同的起始lr, 以及动量