

Neural Networks & Deep Learning¹

—Project2

吴雨涵 21300680121

1、Train a Network on CIFAR-10（代码实现在 MyFramework.py 中）

本模型构建以 Google Inception（v1）为 Baseline 模型²，在其上增加功能与对比，图 1 和图 2 是采用学习率 0.0001，Adam optimizer 训练 200epoch 的结果：

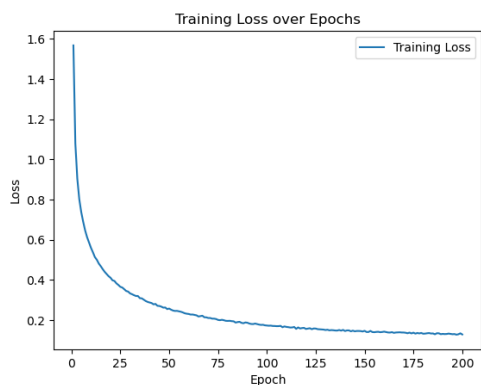


Figure 2

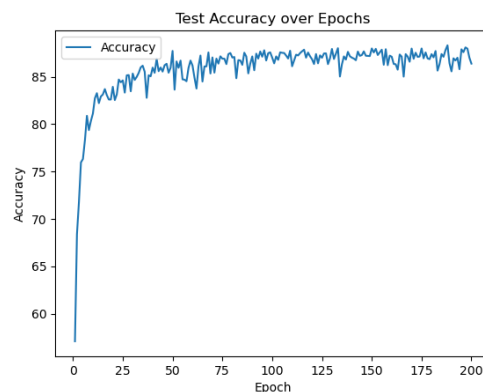


Figure 1

训练时长约为 1h（机器为 RTX A6000），最终准确率稳定在 85% 左右。

改进 1：修改 Inception 块，增加残差连接（如图 3 所示），同时在残差中加入缩放模块（参考 InceptionV4 论文，若不加入缩放训练结果不稳定）

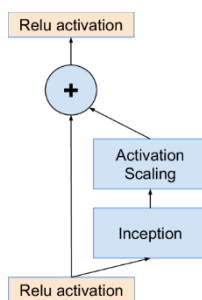


Figure 3

学习率等超参数同上，训练结果见图 4、5：

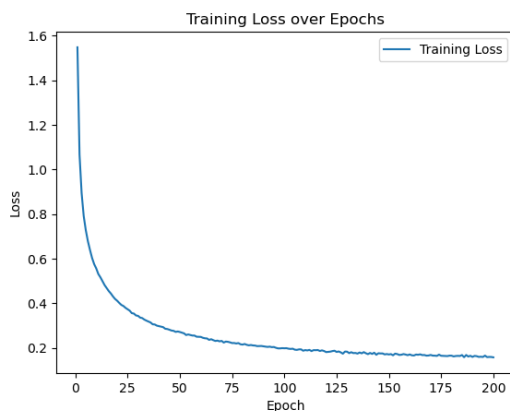


Figure 5

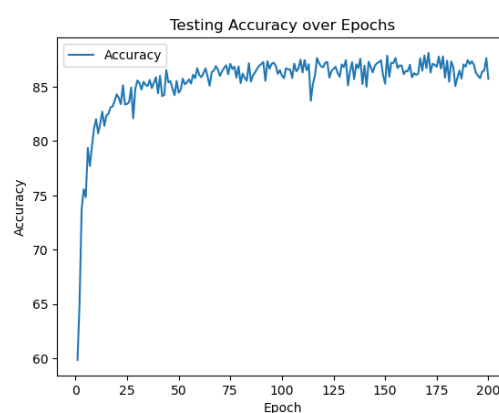


Figure 4

¹ 本项目所有代码均同步于 <https://github.com/Attisu/Project2-for-neural-network> 仓库中

² <https://nndl.github.io/v/cnn-googlenet>，不包含辅助分类器

准确率稳定在 87% 作用，有所上升，但是观察图像可以发现，虽然图像训练到 200epoch 之后 loss 依然在下降，但是 Accuracy 在 50epoch 左右就已经稳定在峰值，可以判断过拟合情况较为严重。

改进 2：使用不同的优化器，添加 L2 正则化 (λ 选用 0.0001)，epoch 数量拓展到 300

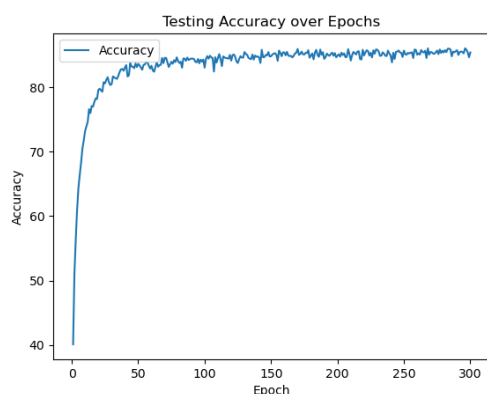


Figure 6 Adam

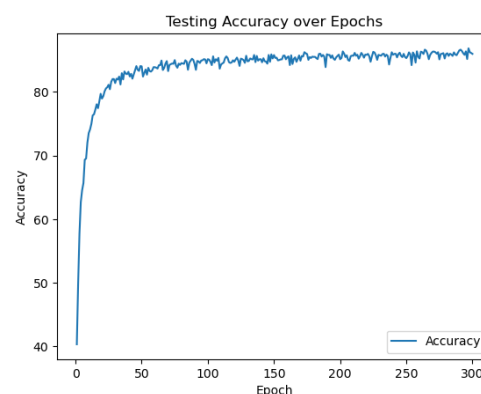


Figure 7 AdamW

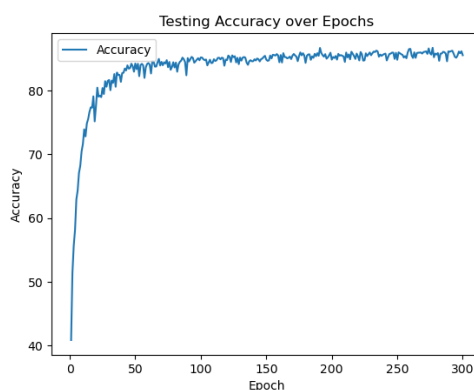


Figure 8 RMSprop alpha=0.99

不同优化器之间的性能表现差异并不大，添加正则化之后普遍 Accuracy 仍然维持在 87% 左右，但是值得注意的是 Accuracy 的波动率显著降低了（相较 Figure 4）

注：模型固定选定 Learning Rate 为 0.0001，不使用动量更新梯度，如果选择更大的学习率或动量更新均会导致梯度爆炸，训练结果不稳定，因此不展现其不同学习率下的损失景观。

2、VGG and Batch Normalization(代码实现在 MyVGG.py 中)

VGG 和添加了批量归一化的 VGG_BatchNorm 的训练结果如下：

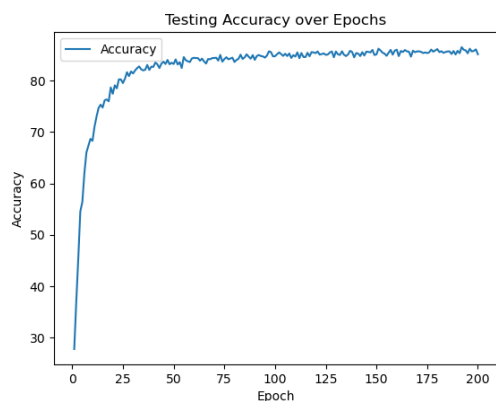


Figure 10 VGG

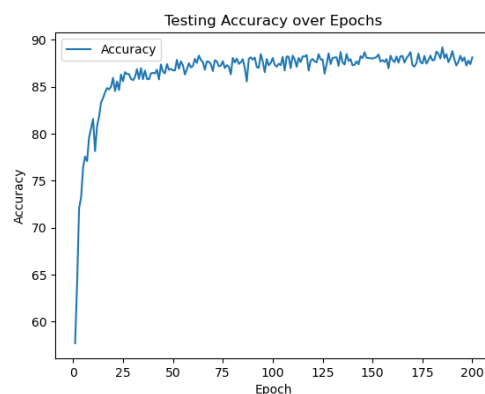
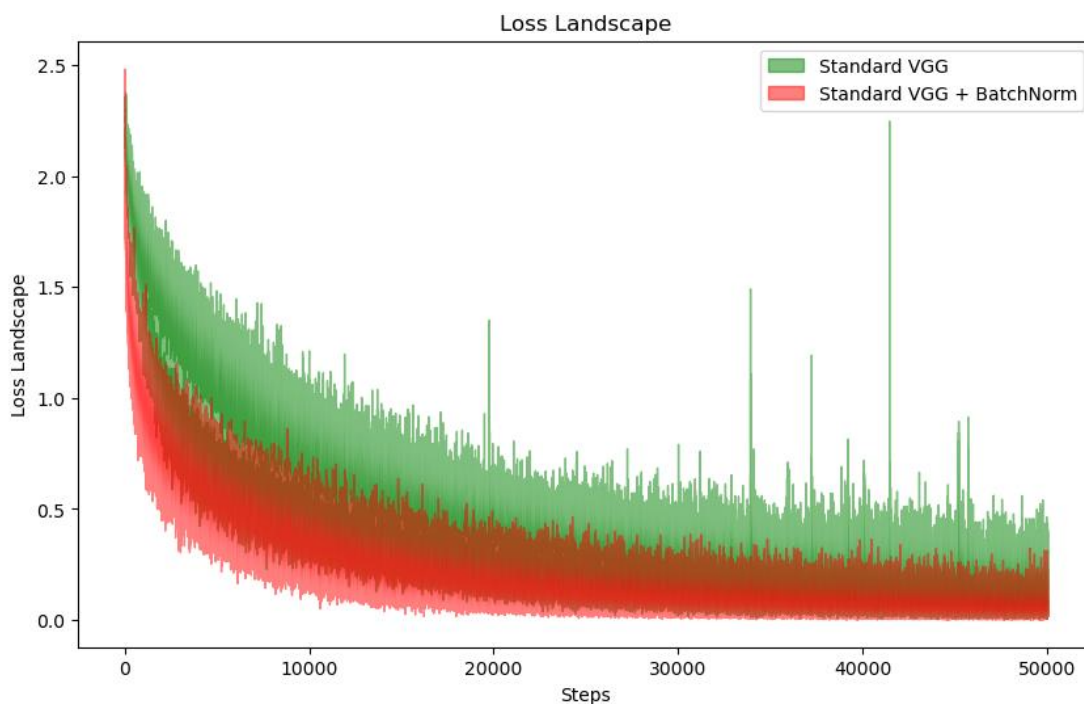


Figure 9 VGG_BatchNorm

在 VGG 基础上，每一次卷积层之后增加一个批量归一化层，训练准确性获得明显提升，由 85%提升到 88%，但是相对而言波动性较强（二者均选择学习率 0.001，L2 正则化系数 0.0001），Loss Landscape（学习率列表为[1e-3, 1e-4, 5e-4, 5e-5]）见下图：



增加批量归一化之后，误差下降明显更快，同时分布更加集中，更少的出现梯度突然升高情况，而在梯度的差异（图中色块的宽度表示）也明显更小，训练结果更稳定，表现也更好。