

# Imagenette 图像识别实验报告

马江岩

2021 年 4 月 11 日

## 摘要

在本实验报告中, 我将介绍我的实验配置、实验结果, 探索不同因素对模型准确率的影响, 并给出自己的分析。经过训练, 我的模型在 imagenette 测试集上达到的最高准确率为 94.5223%。

## 1 实验配置

在实验过程中, 我尝试了不同的优化器、模型和超参数, 以达到更高的测试集识别准确率。这里仅介绍我训练的准确率最高的模型所用的实验配置。

训练该模型时, 我所用的批尺寸 (batch size) 为 64, 优化器 (optimizer) 为 ranger 优化器<sup>1</sup>, 学习率 (learning rate) 为  $8 \times 10^{-3}$ , eps 为  $1 \times 10^{-6}$ , 所用模型为 18 层的残差神经网络 (ResNet)。所用训练集为完整大小的 imagenette 训练集<sup>2</sup>, 训练过程在 Google Colab<sup>3</sup>上完成。

用于训练模型的 Python 代码见附件 Training Model.py, 训练出的模型为附件 ResNet18 0.945223.pth, 用于测试所训练模型的 Python 代码见附件 Testing Model.py。在运行附件中的 Python 代码之前, 需要先下载完整大小的 imagenette 数据集和 ranger 优化器, 并将附件中 Python 代码中的路径改成本地的相应路径。

## 2 实验结果

在不同数量的 epoch 内, 我的模型所达到的最高的测试集识别准确率如表 1 所示。

最终, 在第 526 代, 模型达到了最高准确率 94.5223%。

---

<sup>1</sup><https://github.com/lessw2020/Ranger-Deep-Learning-Optimizer>

<sup>2</sup><https://github.com/fastai/imagenette>

<sup>3</sup><https://colab.research.google.com/>

表 1: 在不同数量 epoch 内模型的最高准确率。

epochs	top1 accuracy
143	92.6624%
203	93.1720%
242	93.6306%
344	93.7580%
385	93.8089%
467	94.1656%
526	94.5223%

### 3 模型分析

这里，我简要分析优化器（optimizer）、学习率（learning rate）和动量（momentum）三个因素对模型效果的影响，并提出我对实验数据的个人解读。

#### 3.1 优化器

在实验初期，我选择的是 SGD 优化器，并按照第 1 章中的超参数进行训练，效果并不理想。后来，我尝试了不同的优化器，最终发现在 imagenette 图像识别任务上，ranger 优化器的效果最好。在学习率为  $8 \times 10^{-3}$ 、批尺寸为 64 的条件下，不同的优化器在 20 代之内的测试集准确率变化曲线如图 1 所示。

可以看出，在 imagenette 图像识别任务上，不同的优化器的表现是不同的。SGD 优化器计算梯度速度快，在处理较大数据集时收敛较快，但在 imagenette 这种较小的数据集上，其收敛速度劣于 ranger 优化器。而 ranger 优化器通过整流函数、自适应动量等机制，可减少超参数调优的工作量，在本任务中达到更快的收敛速度。

当然，图 1 中不同优化器的差距，除了与数据集有关，更多的是受到学习率的影响。 $8 \times 10^{-3}$  的学习率对于 ranger 优化器来说正合适，但对于 SGD 这样下降曲线较为曲折的优化器来说，显然太小了。而图 1 中 Adam 优化器的准确率曲线起伏很大，说明  $8 \times 10^{-3}$  的学习率对 Adam 优化器来说反而偏大了。一般来说，SGD 优化器的默认学习率为 0.1，Adam 优化器的默认学习率为  $1 \times 10^{-3}$ 。

我们不妨将学习率调整为 0.1，同时设置 ranger 优化器的 eps 为  $1 \times 10^{-6}$ 、SGD 优化器的动量为 0.9（根据经验，这些超参数对 ranger 优化器和 SGD 优化器是比较合适的），再来比较 ranger 优化器和 SGD 优化器在 20 代之内的准确率。结果如图 2 所示。

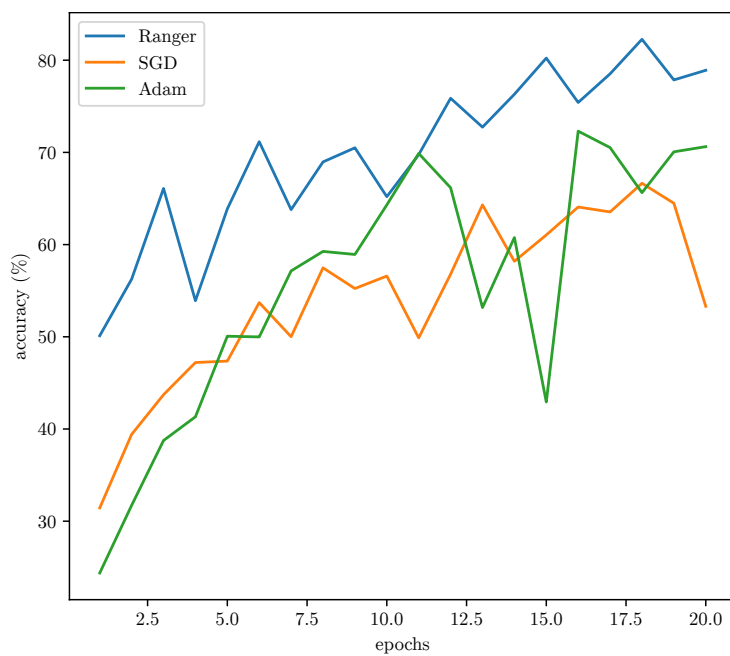


图 1: 学习率为  $8 \times 10^{-3}$  条件下, 使用不同优化器模型的准确率变化曲线。

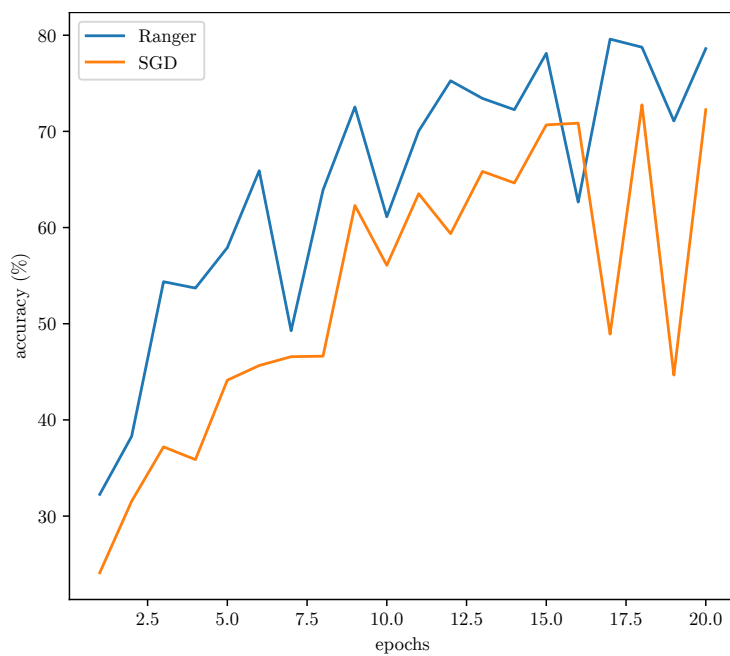


图 2: 学习率为 0.1 条件下, 使用不同优化器模型的准确率变化曲线。

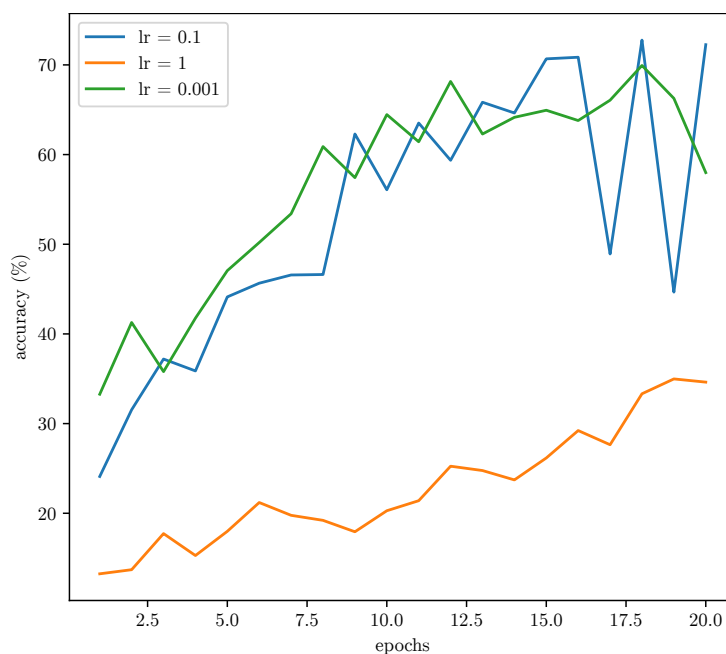


图 3: 使用 SGD 优化器的模型在不同学习率下的准确率变化曲线。

可以看出，在 0.1 的学习率下，使用 SGD 优化器的模型的准确率有所提升，但与使用 ranger 优化器的模型仍有差距。

### 3.2 学习率

本节我将探究学习率对模型准确率的影响。我采用 SGD 优化器和 ResNet-18 模型，在动量为 0.9、批尺寸为 64 的条件下，分别调整 SGD 优化器的学习率为 1、0.1、0.001，将模型在 20 代之内的测试集准确率绘成图表，如图 3 所示。

从图 3 中可以看出，1 的学习率对于 SGD 优化器显然过大，模型准确率低、收敛慢。这是因为，过大的学习率会导致模型沿梯度方向前进距离过大，难以收敛到损失函数的最低点，甚至向使损失函数更大的方向变化。比较 0.1 和 0.001 的学习率下模型的表现，我们可以看出，在 0.1 的学习率下，模型收敛更快，但起伏也更大；在 0.001 的学习率下，模型准确率上升则比较平稳。这是因为，较小的学习率，可以保证模型一步步平稳地向损失函数最低点逼近，不会错过最低点，但收敛速度也更慢。同时，较小的学习率也可能会导致模型陷入局部最优解，从长期来看表现可能不如更大的学习率。

因此，我们要选择适合优化器的学习率，不应过大也不应过小。在实际操作中，我们也可以使用动态学习率规划器 (scheduler) 来控制学习率的动态变化，在训练初期，选择较大的学

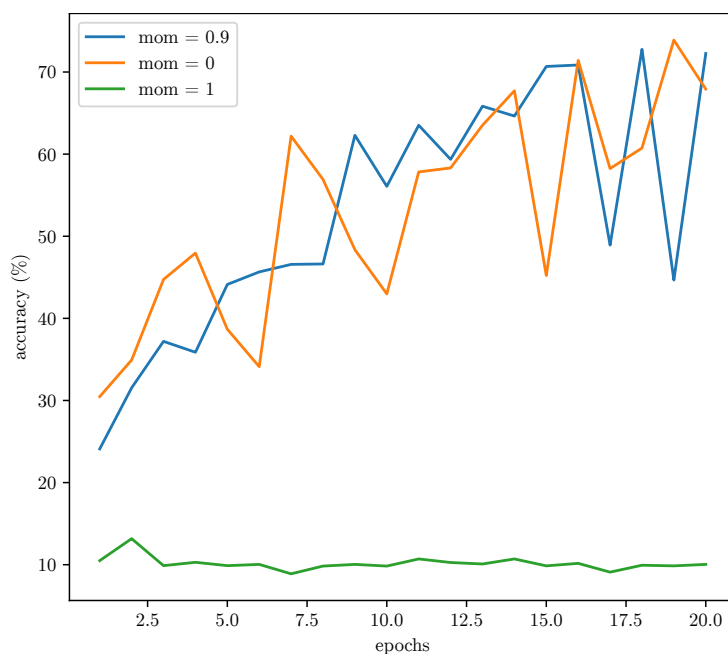


图 4: 使用 SGD 优化器的模型在不同动量下的准确率变化曲线。

习率，使模型收敛到最优解的速度更快，同时减少陷入局部最优的可能性；在后期则降低学习率，使模型平稳地收敛到所在范围内的损失函数最小值点。

### 3.3 动量

本节探究动量对模型准确率的影响。我采用 SGD 优化器和 ResNet-18 模型，在学习率为 0.1、批尺寸为 64 的条件下，分别调整 SGD 优化器的动量为 1、0.9、0，将模型在 20 代之内的测试集准确率绘成图表，如图 4 所示。

从图 4 中可以看出，在 1 的动量下，模型的准确率在 10% 附近震荡，几乎和随机选取无异。这是因为，过大的动量会使模型参数向着一个方向变化，无法下降到损失函数的最小值点，而是四处游荡。而在 0.9 和 0 的动量下，模型的准确率并无明显的差异。我认为，原因在于训练的代数过少，模型还未收敛到最小值，故还无法体现出动量在此处的作用。加入动量可以增强模型的惯性，使模型参数沿着梯度下降幅度较大的方向变化，减少陷入局部最小值的概率。如果训练足够多的代数，具有适当动量的模型应当更可能获得更高的准确率。但在 imagenette 数据集的训练上，经过 200 个 epoch 的训练，动量为 0.9 和 0 的模型在准确率上都未体现出明显差异。或许这是 ResNet-18 模型层数多、参数数量大，我训练的 epoch 未能使模型收敛到最小值点的缘故。

## 4 总结与反思

在本章中，我将简要总结自己在本次实验过程中的成功之处和不足之处，并为今后的训练总结经验教训。

这次实验最终能达到一个比较满意的准确率，一方面是选择了较合适的模型，另一方面是选择了较合适的优化器。由于各种优化器的适宜的超参数在网上都有相关经验介绍，在 imagenette 的 GitHub 网站上也给出了训练准确率较高的模型所用的超参数，因此我在调整超参数上并没有遇到太大的困难。我在实验开始时，为了提高模型的表达能力，选择了层数较多的 ResNet-34 和 ResNet-50 进行训练。这些模型虽然准确率能够较为平稳地上升，但提升速度很慢，即使经过几百个 epoch，效果仍不理想。后来我意识到，模型并不是越复杂效果就越好的，对于 imagenette 这样规模不大的数据集，使用复杂的模型收敛速度会很慢，效果也不如层数更浅的网络。后来我改用了 ResNet-18 进行训练，效果有所提升。我在实验前期使用的是 SGD 优化器，但事实证明，对于规模较小的数据集，SGD 优化器并不能很好地发挥出自己的优势。使用 SGD 优化器，我最高只得到了 89.5% 的准确率。后来我换用 ranger 优化器，按照 imagenette 的 GitHub 主页上给出的超参数进行训练，达到了第 2 章所述的效果。

实验中，我的不足之处主要在于不懂得根据数据集的特点和不同的优化器，选择适宜的模型和超参数。例如，在实验开始阶段，我选择的神经网络要么过于简单，表达能力差；要么过于复杂，训练速度慢。这导致我浪费了大量的时间和计算资源。而当 SGD 优化器的训练效果不理想时，我并没有第一时间考虑更换优化器，而是不断调整超参数，尝试不同的动态学习率规划器，甚至照搬 imagenette 主页上的超参数。实际上，不同的优化器，最优的超参数数值有着很大的不同。我们应当根据优化器的特点和任务的特点，选择不同的优化器。

在之后的神经网络训练中，我将首先根据任务特点，通过尝试和比较，选择合适的神经网络和优化器，避免浪费大量的时间和计算资源。可以根据网上其他人的经验，选择适宜的超参数。还要阅读一些专业论文，提升知识与认识，了解神经网络训练的深层次的原理，以便对训练过程中出现的现象做出合理的判断，并采取相应的改进措施。