

ResNet-18 学习笔记

原理理解

当网络达到一定深度后，随着网络深度增加，浅层网络参数的梯度要经过更多的传递，若梯度值小于1，**梯度消失**；若梯度值大于1，**梯度爆炸**，这都使得深层网络难以训练。

神经网络训练时，可以简单理解为达到两个目标。

- 1. retain the input signal
 - 2. add to the input image
- 前者要求模型学会**恒等映射**，即 x ，后者要求对输入的 x 进行修改，即**残差部分** $F(x)$ 。
而**Resnet**网络通过**残差连接 (Shortcut Connection)**，让更深的网络取得了更好的表现。
$$H(x) = F(x) + x$$

但residual block的设计又存在新的问题需要解决：匹配维度。
此处的维度既是指 height(高度), weight(宽度)，也指channels(频道数量)。
论文中提出了两种方案：**零填充**(zero-padding)和 **1×1卷积**(1×1 convolutions)

Resnet-18/34

结构理解

输入假定为 224×224×3

模块	结构	输出尺寸
Stem	7×7 conv, 64, s=2 → BN → ReLU → 3×3 maxpool, s=2	56×56×64
Conv2	2×BasicBlock (每块: 3×3,64,s=1 → 3×3,64,s=1; 捷径为 identity)	56×56×64
Conv3	2×BasicBlock; 第1块首个3×3 用 s=2 下采样 ，通道到 128；捷径用 1×1,128,s=2 ；第2块全 s=1	28×28×128
Conv4	同上规则：第1块下采样到 14×14×256，捷径 1×1,256,s=2；第2块保持	14×14×256
Conv5	同上规则：第1块下采样到 7×7×512，捷径 1×1,512,s=2；第2块保持	7×7×512
Head	GAP (7×7 → 1×1) → FC → Softmax	num_classes

规律：从 **Conv3** 开始，每个 stage 的**第1个 block 负责下采样 + 通道加倍**；该 block 的捷径用 **1×1 stride=2**。同一 stage 的**后续 block全部 s=1** 保持尺寸不变。

- **初始卷积层**
缩小特征图尺寸
- **四个残差阶段 (Residual Stages)**
Residual Block
每次输出都是特征图的高和宽减半，通道数加倍
- **全局平均池化+全连接层 (FC)**
全局平均池化层来抽取特征
全连接层+softmax函数激活进行分类

张量形状变化

```
1 224→(7×7 s=2)→112 →(maxpool s=2)→56
2 → Conv2: 56×56×64
3 → Conv3: 28×28×128
4 → Conv4: 14×14×256
5 → Conv5: 7×7×512
6 → GAP → 1×512 → FC
```

BasicBlock

- 主分支: $3\times 3, C, s\in\{1,2\} \rightarrow \text{BN} \rightarrow \text{ReLU} \rightarrow 3\times 3, C, s=1 \rightarrow \text{BN}$ 。
- 残差相加后再 ReLU。
- 捷径:
 - 若输入与输出 **(H,W,C)** 一致: Identity。
 - 若 **下采样或通道变化**: 使用**1×1卷积核**进行维度匹配。

Resnet-50/101

结构理解

输入假定为 224×224×3

模块	ResNet-50结构	输出尺寸
Stem	7×7 conv, 64, s=2 → BN → ReLU → 3×3 maxpool, s=2	56×56×64
Conv2	3×Bottleneck ([64,64,256]×3)	56×56×256
Conv3	4×Bottleneck ([128,128,512]×4; 第1块下采样)	28×28×512
Conv4	6×Bottleneck ([256,256,1024]×6; 第1块下采样)	14×14×1024
Conv5	3×Bottleneck ([512,512,2048]×3; 第1块下采样)	7×7×2048
Head	GAP → FC → Softmax	num_classes

Bottleneck

Bottleneck是ResNet-50/101/152等深层网络的核心构建块，通过"先降维-再卷积-再升维"的策略大幅减少参数量。

结构组成

三卷积层设计

- 1×1卷积 (降维)**：将输入通道数压缩到目标通道数，减少后续卷积的计算成本
- 3×3卷积 (特征提取)**：在压缩后的维度上进行常规卷积
- 1×1卷积 (升维)**：将通道数恢复到扩展后的目标维度 (usually expansion=4)

数学表达

1 | 输入x → 1x1卷积 → BN → ReLU → 3x3卷积 → BN → ReLU → 1x1卷积 → BN → +捷径 → ReLU

维度变化

1 | 输入: 256通道 → 1x1卷积 → 64通道 → 3x3卷积 → 64通道 → 1x1卷积 → 256通道

训练时的一些tricks

数据集划分

以 `valid_ratio=0.2` 将5w个训练集划分为训练集和验证集(valid_set)。

基于模型在验证集上的表现, 进行调参。

图像增强

通过一系列变换方法, 在保持图像语义(表达内容)不变的前提下, 生成多样化的新图像数据。以此来增强数据多样性, 提高模型鲁棒性, 减轻过拟合。

```
transforms.RandomCrop(32, padding=4) # 随机裁剪
```

```
transforms.RandomHorizontalFlip() # 50%概率水平翻转
```

Attention

图像增强只能用于训练集 `data_set`, 让模型学到更广泛的特征

随机种子和早停机制

- 随机种子的设置
 - 确保实验的可重复性
- 早停机制
 - **Early Stopping**
 - 在模型即将过拟合(Overfitting)但还未彻底过拟合之前, 提前终止训练
 - 如果性能在连续多个Epoch内不再提升则停止训练, 并保存性能最好的Epoch模型状态

学习率调度

在训练过程中**动态调整学习率**来优化模型的收敛过程和最终性能。

```
scheduler = torch.optim.lr_scheduler.StepLR(optimizer, step_size=30, gamma=0.1)
```

采用**StepLr**的方法, 有效使得epoch次数增加(训练深度增加), 且在测试集表现:

```
test_accuracy=93.88%
```

疑问与解答

1. 如何理解残差连接结构? 它是如何解决梯度消失或爆炸的问题的?
让梯度通过捷径直接传递到给网络的浅层, 避免了梯度层层传递, 带来的导数连乘让监督信号变弱。
2. 不同卷积层的堆叠方式构造出Resnet-18, Resnet-34, Resnet-50模型
 - ResNet-18, ResNet-34比较类似, 只是不同阶段里边的Residual Block个数不同, 但是每个阶段的Residual Block都是一样的。

- ResNet-50, ResNet-101, ResNet-152比较类似，也是每个阶段的Residual Block一样，只是个数不同。

3. BatchNorm 在ResNet中的作用

- 在深度网络中，前一层的参数更新会改变后续层的输入分布，BatchNorm通过**标准化**使每一层的输入分布保持稳定
- 确保梯度在残差块间稳定传递
- 在训练过程中引入噪声

4. 如何解决Conv3-Conv5捷径连接的维度不匹配问题？

进行**下采样**。使用stride=2, 1×1 卷积核来调节特征图尺寸。**那这样不会跳过大量的图像信息吗？**事实上，这样确实会丢失信息，但是残差连接必须保证特征图的长宽，以及通道数都一致才可以进行按位相加。而 1×1 的卷积，步长为2，是达到这一点(长宽减半，通道数加倍)计算量最小的实现。此时信息还是主要靠主通道进行传递。

5. BasicBlock和Bottleneck是如何选择的，为什么要这么设计？

- BasicBlock for ResNet-18/34
- Bottleneck for ResNet-50/101/152
- 为何不继续沿用BasicBlock？减少网络的计算量，让训练更深层的卷积神经网络变得可以承受。