

Pie-Lab 2025年暑期培训

李秋予

Pie-Lab

北京理工大学计算机学院

北京市海淀区中关村南大街

786362411@qq.com

Abstract

以下是CIFAR-10图像分类的实验报告，包括网络结构的原理解释、模型结构设计与损失函数选择、实验设置、实验结果与可视化和总结与分析。

1 网络结构的原理解释

1.1 普通CNN模型(Convolutional Neural Network)

卷积神经网络是一类包含卷积计算且具有深度结构的前馈神经网络(Feedforward Neural Networks),它通过卷积操作来提取输入数据的局部特征，并通过多层卷积和池化操作形成复杂的特征表示，最终通过全连接层进行分类或回归等任务。

1.输入层(Input Layer):

在神经网络中，输入层(Input Layer)是神经网络的第一层，负责接收外部输入的数据,这些数据可以是图像、声音、文本、传感器数据等,输入层的主要作用是这些数据以数值的形式传递给网络，以便进行后续的处理和分析。

2.卷积层(Convolution Layer):

卷积层(Convolution Layer)是卷积神经网络(Convolutional Neural Network,CNN)中的核心层(核心组件)之一，它主要负责在输入图像上执行卷积操作，以提取输入图像中的不同特征。卷积层通过卷积核(滤波器或特征检测器)与输入图像进行局部连接和卷积运算，以生成特征图(Feature Map)，这些特征图包含了输入数据的不同特征信息。

(1)卷积核大小f(Kernel Size):定义了卷积操作时覆盖的输入数据区域大小，影响特征提取的尺度。

(2)步长s(Stride):卷积核在输入数据上滑动的距离，影响输出特征图的尺寸。

(3)填充p(Padding):在输入数据边缘添加的额外像素，用于控制输出特征图的尺寸。

(4)输入/输出通道数(In/Out Channels):输入数据的深度/通道数和输出特征图的数量,决定了卷积层处理的深度和能够生成的特征图种类。

输入图片的尺寸用 n 表示,输出图片的尺寸用 o 来表示,可以求得计算公式如下:

$$o = \left\lfloor \frac{n + 2p - f}{s} \right\rfloor + 1$$

其中“ \lfloor ”是向下取整符号,用于结果不是整数时进行向下取整。

3.池化层(Pooling Layer):

池化层(Pooling Layer)是深度学习神经网络中常用的一种层级结构,主要用于减小输入数据的空间尺寸,降低模型的计算复杂度,减少过拟合,并在一定程度上提取输入数据的重要特征。池化层主要对卷积层输出的特征图进行下采样操作,一般跟在卷积层之后。

4.全连接层(Fully Connected Layer):

全连接层(Fully Connected Layer)是深度学习中常用的一种神经网络层,常用于图像识别等,它的主要作用是学习到前面层(如卷积层、池化层等)输出的特征,进行全局整合,并映射到样本的标记空间(输出空间)。

5.ReLU函数

线性整流单元(ReLU), ReLU提供了一种非常简单的非线性变换。给定元素 x , ReLU函数被定义为该元素与0的最大值。

$$ReLU(x) = \max(x, 0)$$

ReLU函数通过将相应的活性值设为0,仅保留正元素并丢弃所有负元素。如下为ReLU函数的曲线图。

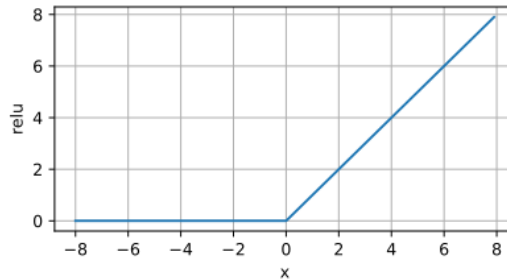


图 1: ReLU函数的曲线图

当输入为负时, ReLU函数的导数为0,而当输入为正时, ReLU函数的导数为1。当输入值等于0时, ReLU函数不可导。如下为ReLU函数的导数:

$$f'(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases}$$

ReLU函数的求导表现的很好:要么让参数消失,要么让参数通过,ReLU减轻了神经网络的梯度消失问题。

1.2 残差网络(ResNet)

残差网络(Residual Network),简称ResNet,是一种深度卷积神经网络(CNN)架构,由微软研究院提出,它在2015年的ImageNet竞赛中获得了冠军。

因为增加网络的宽度和深度可以很好的提高网络的性能,深的网络一般比浅的的网络效果好,因此在训练模型时,会想到加深网络模型的层数,但是这也随即带来梯度消失和退化问题。

1.梯度消失(Gradient Vanishing):在传统的深度神经网络中,梯度在反向传播过程中会逐层递减,导致较深层的权重更新较小。这会导致较深层的网络参数难以得到有效的训练,而浅层的网络参数则可能过度拟合训练数据。梯度消失问题使得网络难以收敛和优化。

2.网络退化(Degradation Problem):随着网络层数的增加,网络的性能反而下降。即使网络的深度增加,其训练误差也会逐渐增加,这与我们期望的随着网络深度增加而获得更好性能的直觉相悖。这种现象被ResNet团队称为网络退化问题,它限制了深层网络的有效性和可扩展性。

因此,为了解决这些问题,ResNet(Residual Network)应运而生。

传统网络:试图学习输入 x 到目标输出 $H(x)$ 的直接映射,即 $H(x) = F(x)$ 。

残差网络:将目标分解为残差 $F(x) = H(x) - x$,网络只需学习残差部分,最终输出为 $H(x) = F(x) + x$ 。

残差块的典型结构(以ResNet中的基本块为例):

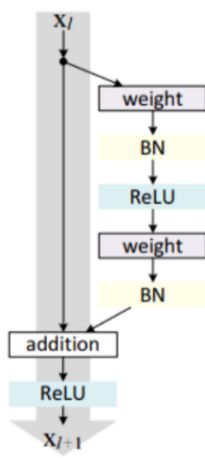


图 2: 残差块

卷积层通常使用3*3卷积核,保持特征图尺寸不变(padding=1),BatchNorm为标准化输入分布,加速训练并增强稳定性,用ReLU引入非线性,增强网络表达能力,最后直接将输入x加到卷积输出上,实现“跳跃连接”。

当残差块的输入与输出通道数或尺寸不一致时,需通过1*1卷积调整捷径连接,改变通道数或特征图尺寸,确保 $F(x)$ 与x形状一致。

2 模型结构设计与损失函数选择

2.1 模型结构设计与参数设置

1.普通卷积神经网络(CNN)

采用“卷积层+池化层+全连接层”的经典架构,具体如下:

输入(3*32*32)->卷积层1->ReLU->池化层->卷积层2->ReLU->池化层->卷积层3->ReLU->池化层->展平->全连接层1->ReLU->全连接层2->ReLU->全连接层3->输出(10类)

(1)卷积层:

通过滑动卷积核提取局部特征,通道数增加代表特征维度提升。

conv1:输入3通道(RGB图像),输出16通道,3*3卷积核(padding=1, stride=1),卷积后尺寸为32*32,保持尺寸32*32

conv2:输入16通道(conv1的输出),输出32通道,卷积后尺寸16*16(池化后)->卷积后仍为16*16

conv3:输入32通道(conv2的输出),输出64通道,卷积后尺寸8*8(池化后)->卷积后仍为8*8

(2)池化层:

降低特征图尺寸,减少计算量,增强平移不变性。

2*2最大池化(stride=2,无重叠采样),每次池化后尺寸缩小为原来的1/2

(3)全连接层:

将卷积层输出的高维特征映射到类别空间,完成分类决策。

fc1:输入:展平后的卷积特征(64通道*4*4尺寸),输出:512维特征

fc2:输入:512维特征,输出:128维特征

fc3:输入:128维特征,输出:10维(对应CIFAR-10的10个类别,最终分类得分)

(4)激活函数:ReLU

引入非线性,解决线性模型表达能力不足的问题;缓解梯度消失,加速收敛。

2.残差网络(ResNet)

针对深层网络的“退化问题”(深度增加但性能下降), ResNet引入残差连接:让网络学习“残差映射”(F(x)=H(x)-x)而非直接映射(H(x))。当H(x)接近恒等映射时, F(x)接近0, 网络更容易优化, 同时梯度可通过捷径直接回传, 解决梯度消失问题。

(1)残差块(ResidualBlock):

双卷积结构:conv1(3*3)->BN->ReLU->conv2(3*3)->BN

残差连接:out=F(x)+shortcut(x), 其中F(x)是卷积输出, shortcut(x)是输入的调整版本
捷径连接(Shortcut):

当stride=1且通道数匹配时, shortcut(x)=x(直接传递)

当stride 1(下采样)或通道数不匹配时, 用1*1卷积调整维度(如nn.Conv2d(in,out,1,stride))

(2)残差模块:

通过多块堆叠加深网络, 同时保持梯度流动顺畅,第一个残差块:可能通过stride=2进行下采样,若输入通道数与输出通道数不匹配,则使用1*1卷积调整捷径连接的维度,后续残差块:stride=1,不改变特征图尺寸,由于输入输出通道数一致,无需调整捷径连接。

layer1:16通道, 2个残差块(stride=1, 尺寸32*32)

layer2:32通道, 2个残差块(首个块stride=2, 尺寸16*16)

layer3:64通道, 2个残差块(首个块stride=2, 尺寸8*8)

(3)全局平均池化:

将64*8*8的特征图转为64*1*1, 替代全连接层的早期压缩, 减少参数和过拟合。

2.2 损失函数选择

本实验采用交叉熵损失函数(CrossEntropyLoss), 适用于多分类任务, 选择原因如下:

1.函数原理

交叉熵损失衡量预测概率分布与真实分布的差异, 公式为:

$$\text{Loss} = - \sum_{c=1}^C y_c \cdot \log(p_c)$$

$C = 10$ (CIFAR-10类别数), y_c 是真实标签的one-hot编码(仅目标类别为1), p_c 是模型预测的类别概率(通过SoftMax计算), 因为PyTorch的CrossEntropyLoss已内置SoftMax, 直接接收模型输出的logits(未归一化得分), 因此无需手动添加SoftMax层。

3 实验结果与可视化

在相同训练环境下，CNN最终准确率达76.72%，最佳准确率出现在第28个epoch；ResNet最终准确率为85.13%，准确率提升8.41%，最佳准确率出现在第30个epoch，残差连接在提升模型性能上有所提升。

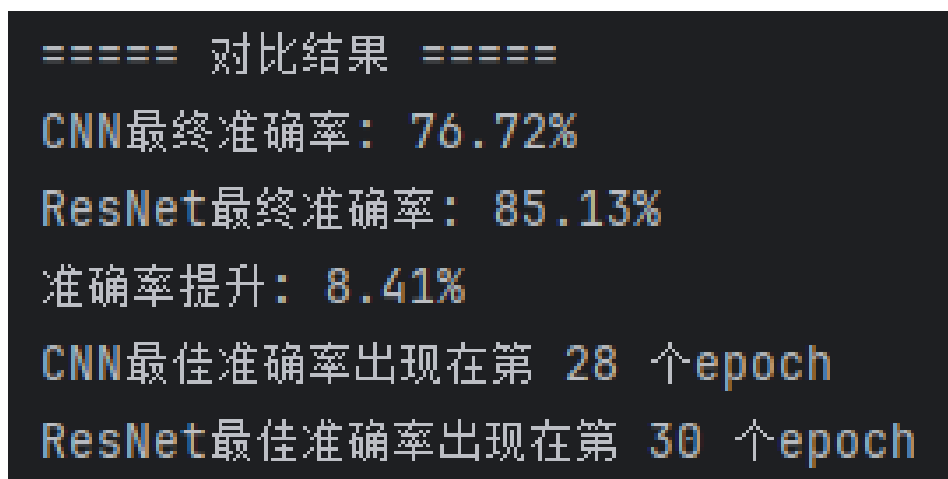


图 3: 结果对比

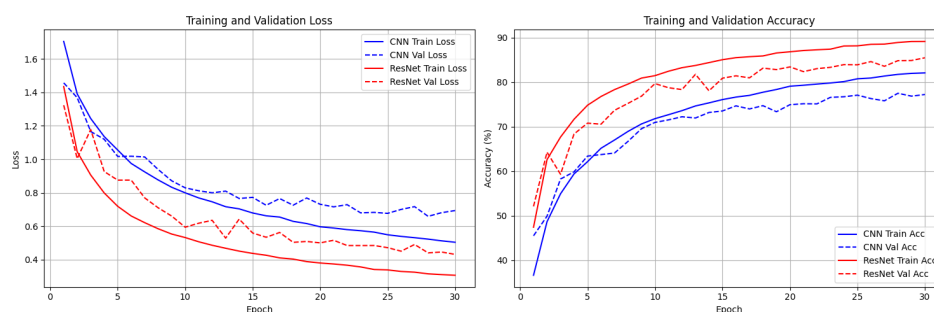


图 4: CNN与ResNet结果可视化

4 总结与分析

在本次围绕CIFAR-10数据集开展的图像分类实验中，我对比了传统卷积神经网络（CNN）与融入残差连接的ResNet网络的性能表现，探究了不同网络结构及训练机制对模型精度与训练效率的影响。

从模型精度来看，普通CNN最终达到了76.72%的准确率，而ResNet则将准确率提升至85.13%，提升幅度为8.41%。这一结果显示了残差连接在增强网络学习能力方面的效果。残差连接通过引入捷径连接，允许网络学习残差映射，避免了传统深层网络中普遍存在的梯度消失问题，使得深层网络参数更新得以顺利进行，进而捕捉到更为丰富、高阶的图像特征，提升分类准确性。

训练过程中，CNN与ResNet在收敛特性上呈现出差异。CNN的训练损失下降过程较ResNet来说更为缓慢，这表明其在优化过程中面临梯度不稳定、收敛困难等问题。在准确率提升方面，ResNet同样展现出优势，在训练初期即可提升准确率，并在后续训练中持续优化。

残差连接虽为ResNet带来了性能提升，但也在一定程度上增加了单次迭代的计算开销。在每个残差块中，逐元素相加操作以及可能出现的 1×1 卷积维度调整操作，导致ResNet的单次迭代时间比普通CNN略长。然而，由于收敛速度的提升，ResNet实现了以单次计算代价换取整体训练效率的提升。

综合来看，残差连接是一种有效的网络设计策略，能够提升模型的训练效果与泛化能力。