

LeNet-5在MNIST上训练与测试

技术报告

|  |  |  |  |
| --- | --- | --- | --- |
| **题 目:** | LeNet-5在MNIST上训练与测试 | | |
| **姓 名:** | 安家琪 | **学 号：** | 122106222787 |

2023 年 4 月

**1.1实验要求**

实现LeNet-5在MNIST数据集上的训练与测试，并进行结果分析。

**1.2 数据集MNIST简介**

MNIST数据集来自美国国家标准与技术研究所，National Institute Standards and Technology（NIST）。训练集（training set）来自250个不同人手写的数字构成，其中50%是高中生，50%来自人口普查局（the Census Bureau）的工作人员。测试集（test set）也是同样比例的手写数字数据，但保证了测试集与训练集的作者集不相交。

MNIST数据集一共有7万张，其中6万张是训练集，1万张是测试集。每张图片是28 \* 28的0-9的手写数字图片组成。每个图片是黑底白字的形式，黑底用0表示，白底用0-1之间的浮点数表示，越接近1，颜色越白。

实验中，利用pytorch框架的torchvision库下的dataset函数对MNIST数据进行读取，并用torch.utils.data库下的DataLoader函数进行打包，代码如图1。

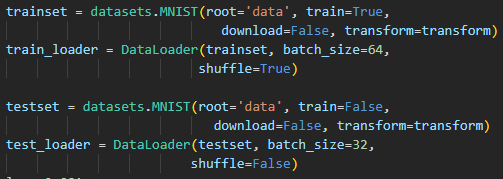


图 1 MNIST数据读取代码

如图2为对MNIST数据集的可视化展示。

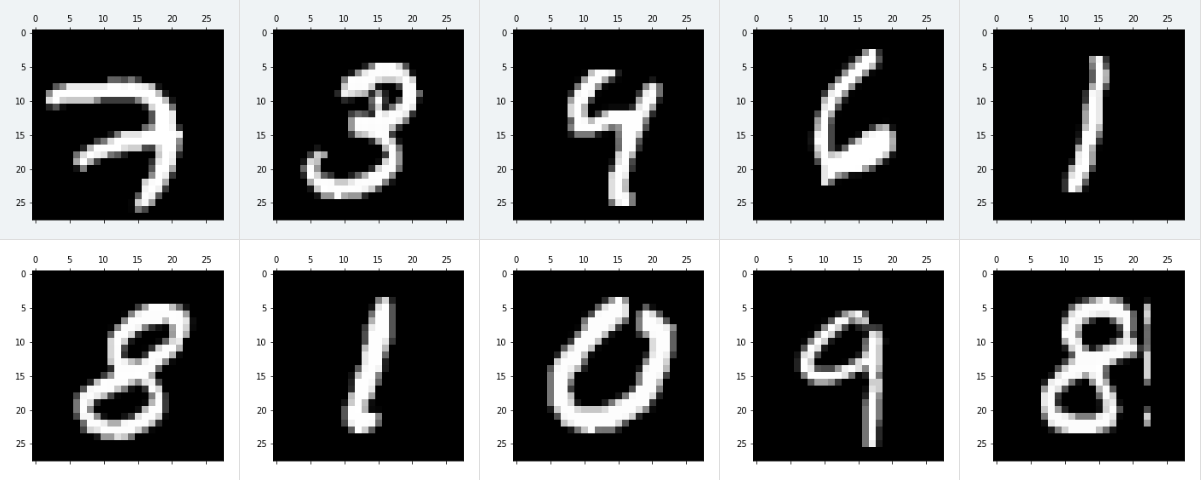


图 2 MNIST数据集可视化展示

**1.3 方法描述 - LeNet-5**

LeNet-5模型是Yann Lecun教授在1988年提出的，成功应用数字识别问题的神经网络。在MNIST数据集上达到了99.27%。

LeNet-5（如图3）一共由7层组成，分别是C1、C3、C5卷积层，S2、S4 降采样层（降采样层又称池化层），F6为一个全连接层，输出是一个高斯连接层，该层使用softmax函数对输出图像进行分类。为了对应模型输入结构，将MNIST中的28 \* 28的图像扩展为32 \* 32像素大小。下面对每一层进行详细介绍。

C1卷积层由6个大小为5 \* 5的不同类型的卷积核组成，卷积核的步长为1，没有零填充，卷积后得到6个28 \* 28像素大小的特征图；S2为最大池化层，池化区域大小为2 \* 2，步长为2，经过S2池化后得到6个14 \* 14像素大小的特征图；C3卷积层由16个大小为5 \* 5 的不同卷积核组成，卷积核的步长为1，没有零填充，卷积后得到16个10 \* 10像素大小的特征图；S4最大池化层，池化区域大小为2 \* 2，步长为2，经过S2池化后得到16个5 \* 5像素大小的特征图；C5卷积层由120个大小为5 \* 5的不同卷积核组成，卷积核的步长为1，没有零填充，卷积后得到120个1 \* 1像素大小的特征图；将120个1 \* 1像素大小的特征图拼接起来作为F6的输入，F6为一个由84个神经元组成的全连接隐藏层，激活函数使用sigmoid函数；最后一层输出层是一个由10个神经元组成的softmax高斯连接层，可以用来做分类任务。

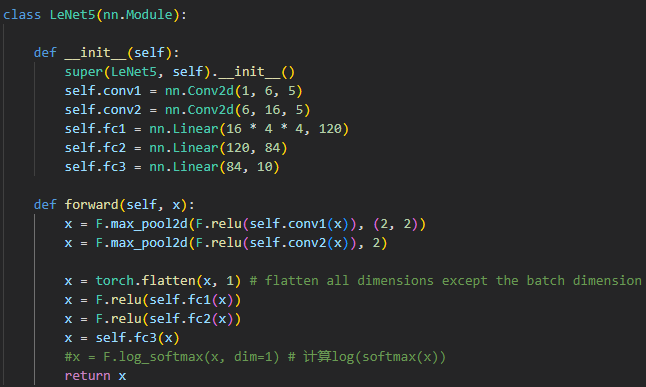


图 3 LeNet-5代码

**1.4 训练结果展示**

如图4为使用LeNet-5网络对MNIST训练集上的训练损失下降图；图5为在训练集上的精确度上升图；图6为LeNet-5网络对MNIST测试集上的损失下降图；图6为在测试集上的精确度上升图。

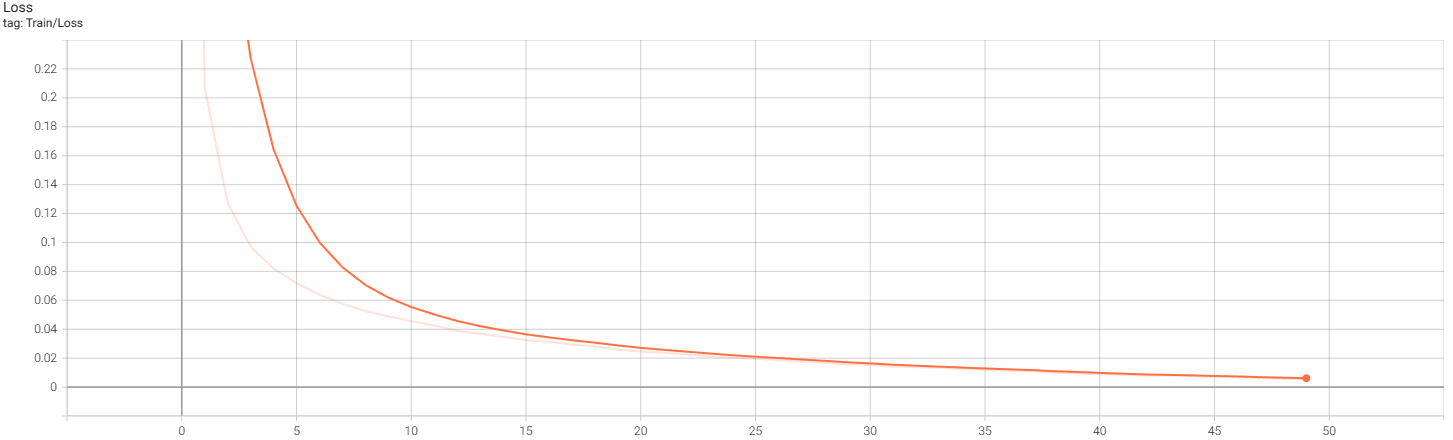


图 4 Train-loss

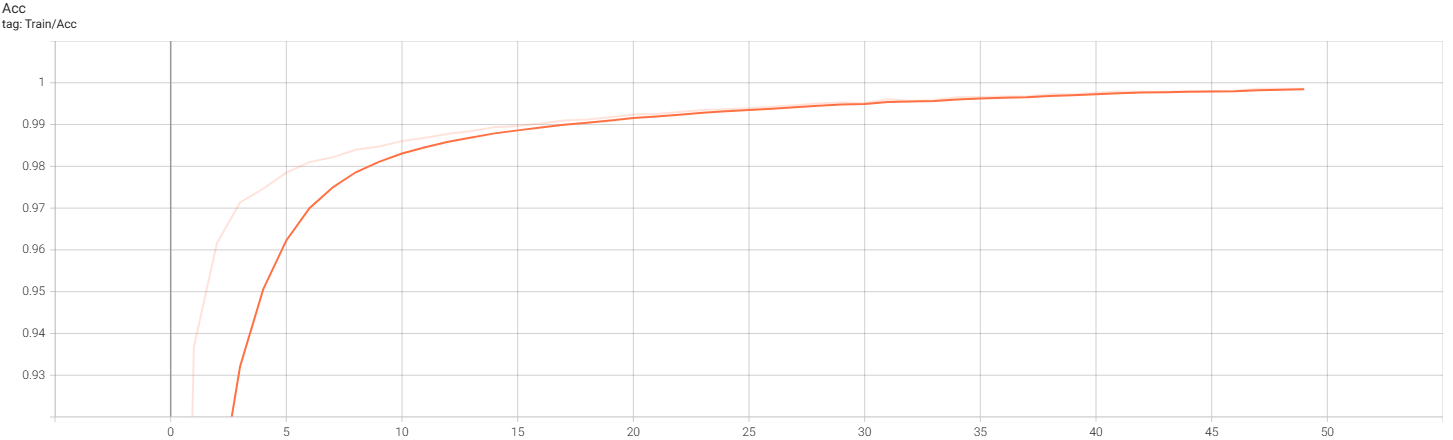


图 5 Train-acc

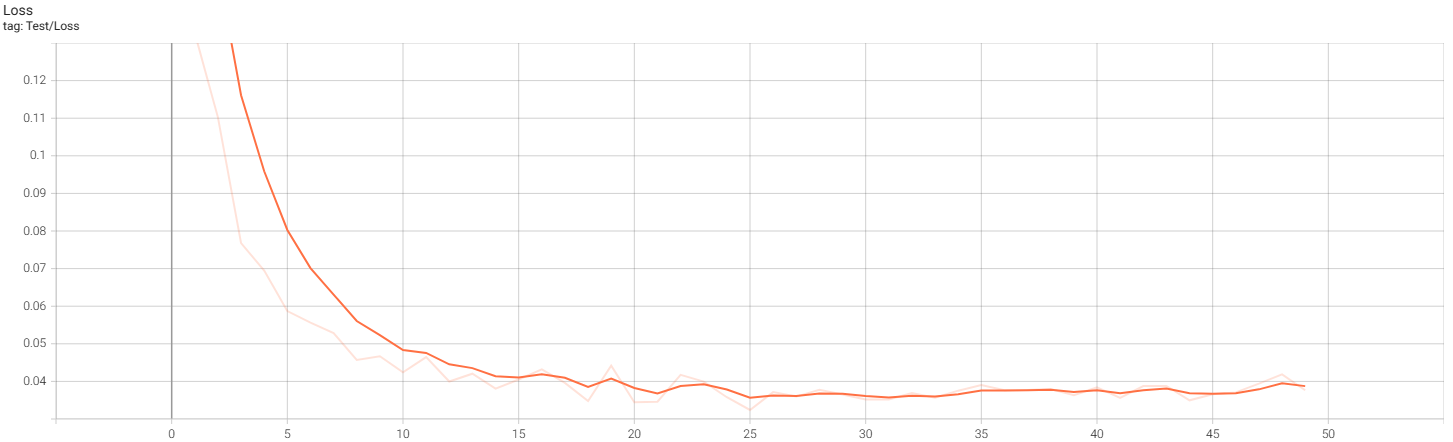


图 6 Test-loss

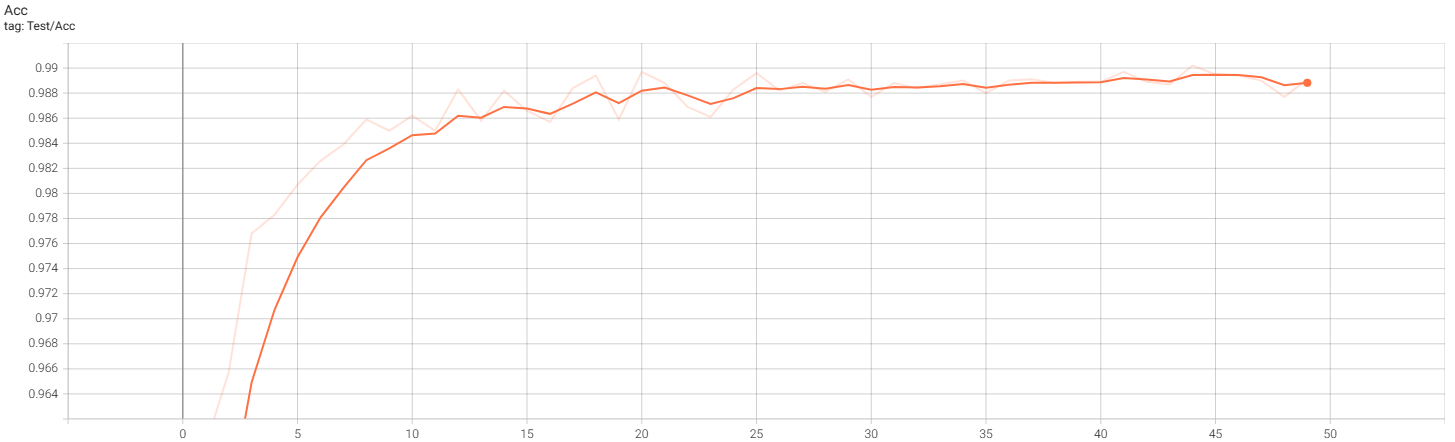


图 7 Test-acc

由上述几幅图可以观察到在训练过程中的训练损失值在不断下降，最终稳定在0.1以下，acc不断上升最终稳定在99.5左右。在测试集上也展示了不错的效果最高的acc值达到了99.02。但相比于训练集要准确度上升的过程中相对产生的波动要更大一些。

**1.4 实验设置**

实验在Pytorch框架下实现。训练期间使用SGD优化器，学习率设置为0.01，使用CrossEntropyLoss进行损失计算，实验进行50轮迭代训练，训练集batchsize设置为64， 测试集设置为32。代码如图8。

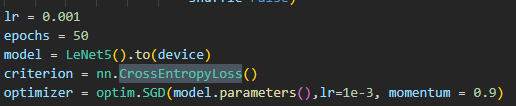


图 8 实验超参数设置

**1.5 实验分析总结**

通过此次实验，深刻掌握了利用深度学习进行实际任务的实现过程。了解了LeNET-5的设计初衷以及构造原理，以及其对分类任务的工作原理。动手复现了模型结构代码并搭建深度学习架构进行对手写数字分类任务的完成。通过对损失值以及精确度的可视化展现了模型对此类任务的极高完成度。通过对模型每一层的挖掘体会到了模型层层之间的联系，促使我期望自己能够搭建自己的模型来完成任务，收获颇大。