### LeNet5网络在MNIST数据集上的训练与测试

1. .实验任务：使用LeNet5网络对MNIST数据集进行手写数字识别
2. .实验框架：使用MindSpore框架进行网络的训练和测试
3. .网络结构：

使用卷积核进行特征提取：

卷积核1：输入通道为1，输出通道为6，步长为1的5\*5卷积核

卷积核2：输入通道为6，输出通道为16，步长为1的5\*5卷积核

使用池化进行下采样：

最大池化：步长为2的2\*2池化核

使用RELU进行非线性激活

神经网络的最终输出为10维向量，通过softmax进行归一化，每个维度代表一个类别评分

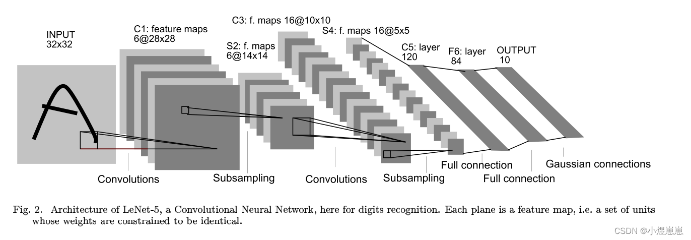


图3-1 LeNet5网络结构

1. .网络训练：
   1. 网络结构搭建：定义网络类继承自mindspore.nn.Cell,在mindspore文档查找相应的卷积池化层，以及非激活函数进行网络搭建。注意：前向网络要重写construct方法
   2. MNIST数据集导入：使用mindspore.dataset库中MnistDataset导入MNIST数据集原始数据，同时将数据集分成训练集和测试集。后使用ms库对数据进行变换将原始数据调整成mindspore网络要求的格式，以便后续训练。做了数据类型变换和图像格式变换（将图像HWC转换为CHW格式）
   3. 损失函数：使用SoftmaxCrossEntropyWithLogits作为函数损失，此函数是将神经网络的输出向量先做softmax再做交叉熵损失。使用此损失后函数可以不在前向网络加softmax归一化
   4. 优化器：使用Adam优化器进行参数优化
   5. 模型封装：使用mindspore库中Model类传入LeNet5网络实例、损失函数、优化器将网络进行封装，后调用Model.train()方法传入epoch和训练集集进行网络训练
   6. 网络参数保存：mindspore.save\_checkpoint()传入网络实例，以及参数保存路径进行参数保存
2. .网络测试：

model.eval()方法传入测试集，返回一个字典，查找accuracy键读出accuracy的值

1. .训练、测试结果：

使用Adam优化器进行优化，训练1 epoch，网络迅速收敛，最终测试准确率为98.03%

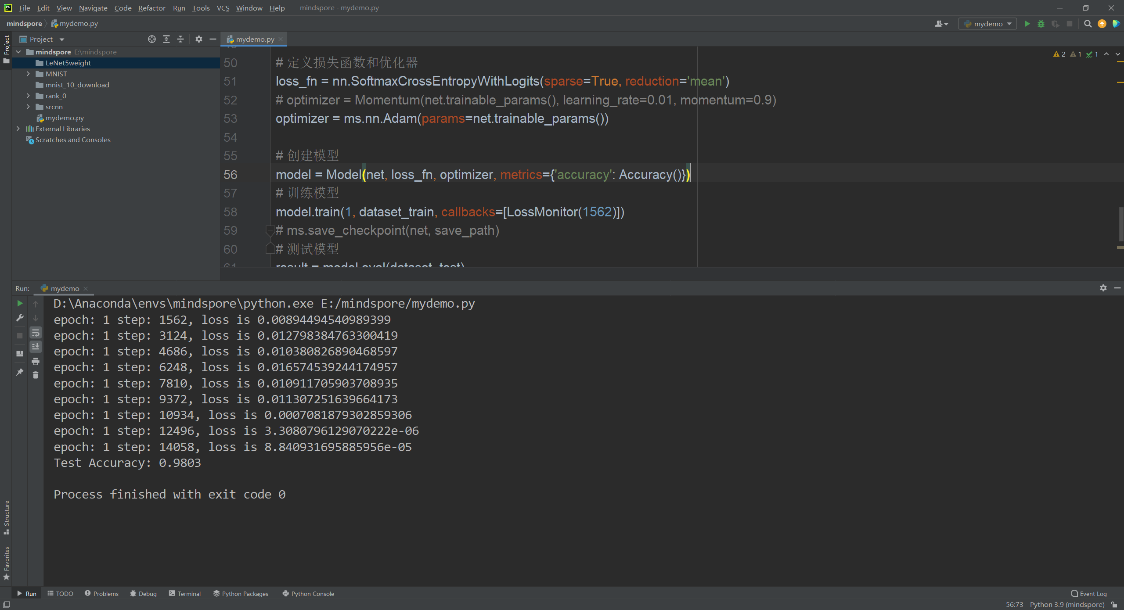


图6-1 LeNet5网络测试结果