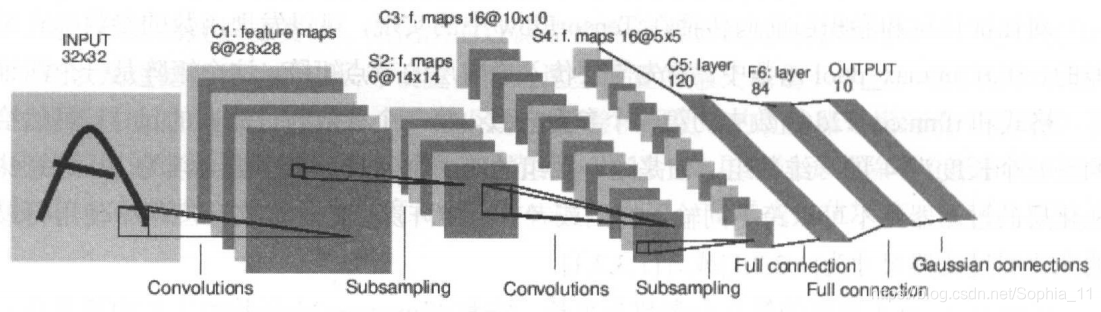
**手写数字体识别实验**

1. **实验目标**
2. 实现LeNet-5在MNIST数据集上的训练和测试，并进行分析。
3. **实验要求**
4. 理解LeNet-5模型的基本原理和结构；
5. 熟悉MNIST数据集，包括数据的获取、预处理和加载；
6. 使用深度学习框架（如MindSpore、TensorFlow或PyTorch）实现LeNet-5模型；
7. 训练LeNet-5模型，并进行必要的超参数调优以提高模型性能。
8. **实验步骤**
9. LeNet-5模型

LeNet由Yann Lecun 提出，是一种经典的卷积神经网络，是现代卷积神经网络的起源之一。Yann将该网络用于邮局的邮政的邮政编码识别，有着良好的学习和识别能力。LeNet又称LeNet-5,具有一个输入层，两个卷积层，两个池化层，3个全连接层（其中最后一个全连接层为输出层）。

LeNet5 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。

以下是LeNet-5模型的结构图：  


1. 具体实现

①**安装依赖**：首先，您使用pip命令安装MindSpore和VisualDL。由于您可能遇到了网络问题，您尝试通过清华大学的TUNA镜像站来加速下载。

②**下载数据集**：使用自定义的download函数从指定的URL下载MNIST数据集，并将其保存在当前目录下。

③**导入必要的库**：导入MindSpore框架的核心库，以及用于数据可视化的VisualDL。

④**初始化日志记录器**：使用VisualDL的LogWriter来记录训练过程中的损失和准确率。

⑤**数据集加载与预处理**：加载MNIST数据集，并定义一个数据处理流水线，包括图像缩放、归一化和转换为CHW格式。

⑥**定义LeNet5模型**：创建一个LeNet5模型，该模型包含两个卷积层、三个全连接层以及ReLU激活函数和最大池化层。

⑦**定义损失函数和优化器**：选择交叉熵损失函数和随机梯度下降优化器。

⑧**定义训练和测试函数**：定义了forward\_fn来计算损失和输出，grad\_fn来计算梯度，train\_step来执行单步训练，以及train和test函数来执行多步训练和测试。

⑨**训练与测试循环**：在指定的轮数（epochs）内，对模型进行训练，并在每个epoch结束后进行一次测试，记录损失和准确率。

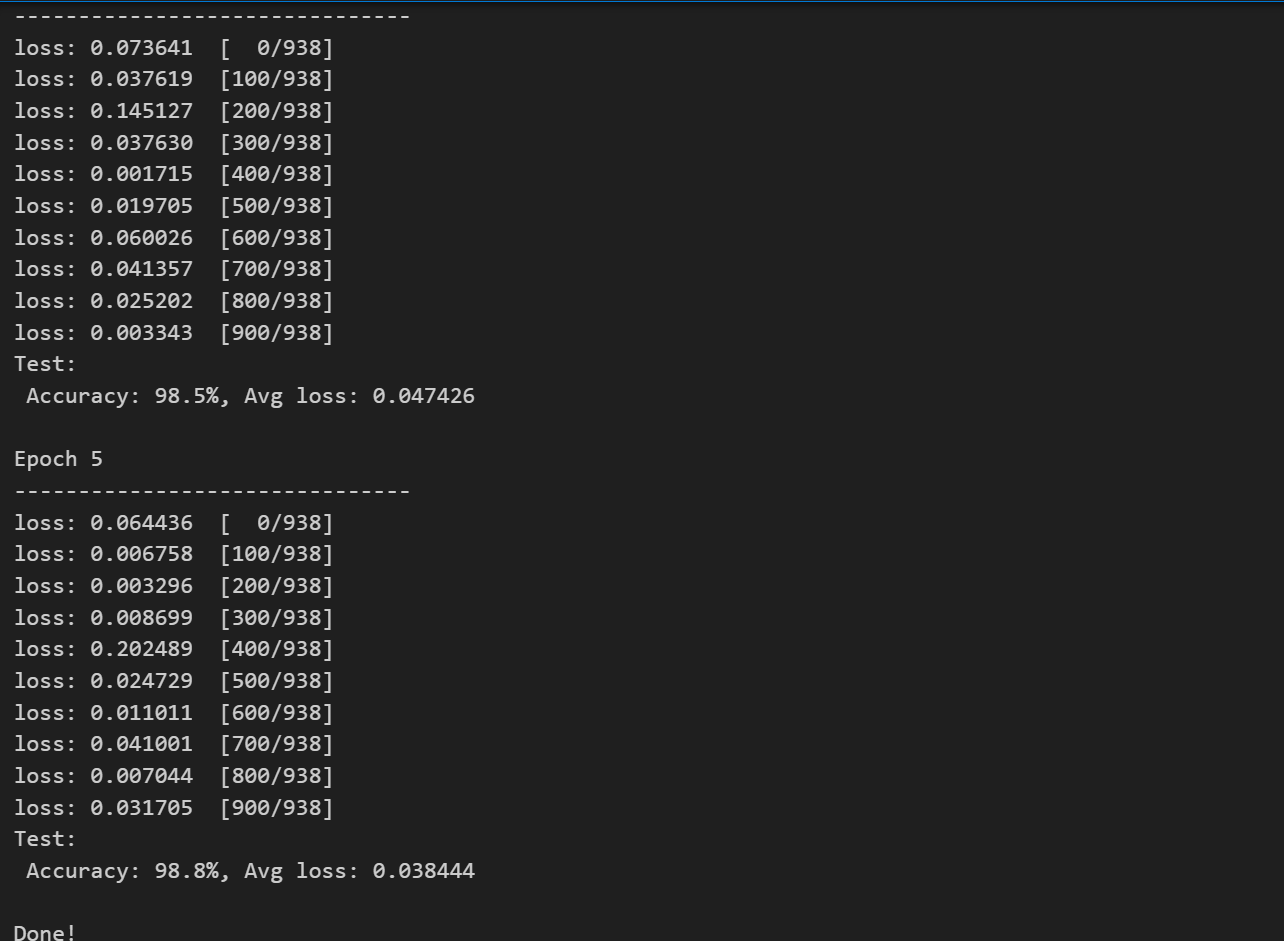
⑩**关闭日志记录器**：在所有训练和测试完成后，关闭日志记录器以保存数据。

⑪**保存模型**。

1. 算法流程



1. **结果与分析**
2. 结果



1. 分析

本次实验我使用LeNet-5模型来进行手写数字识别的实验，经过5个epoch的训练之后测试集的准确率已经达到了98.8%，同时模型也已经收敛，可以认为该模型已经较好地学习了MNIST数据集的特征。

1. **总结**

本次实验聚焦于手写数字识别，使用的数据集是MNIST数据集，并使用经典的LeNet-5模型来进行训练和测试。

在实验的过程中，我使用了Mindspore框架来实现整个算法。我首先安装了MindSpore框架，这是华为推出的开源深度学习框架，它支持多种硬件平台，能够灵活地进行模型训练和推理。通过MindSpore的MnistDataset类，我便捷地加载了MNIST数据集。同时，我利用框架提供的API设计了数据处理流水线，包括图像的缩放、归一化和批处理，确保了数据输入的质量。我选择了LeNet-5作为实验的神经网络模型。在MindSpore中，我通过定义LeNet类，构建了包含卷积层、池化层和全连接层的模型架构，并根据数据预处理对模型结构进行了修改。我使用MindSpore的高级API来定义损失函数、优化器，并实现了模型的训练和测试循环。在训练过程中，我通过VisualDL记录了损失和准确率，这让我能够实时监控模型的学习进度。我利用VisualDL的日志记录功能，定期记录了训练和测试过程中的关键指标，这帮助我分析了模型的收敛情况，并及时调整了训练策略。

通过本次实验，我不仅加深了对深度学习理论的理解，还提升了使用MindSpore框架解决实际问题的能力。我认识到了在模型设计、训练调优和性能评估方面的一些关键点。