

北航计算机学院深度学习系统 课程实验报告

Lab-1：简单神经网络

学 生 学 号：ZY2406334

学 生 姓 名：彭旭桐

二〇二四年九月

摘 要

本实验的主要目标是搭建基本的多层神经网络，并在给定 Mnist 官方数据集上进行精度测试，并要求不使用深度学习框架。

实验过程中使用了 python 的 numpy 库以及 cupy 库进行计算，同时使用了 os 等库进行数据的 I/O 操作。撰写代码主要包括：模型文件、训练文件和工具类文件。

实验结果为两个模型参数 pkl 文件以及它们在测试集上相应的精度。

1 引言

1.1 实验目的

使用全连接层网络和 Mnist 数据集训练出能识别手写数字的模型，并在测试中取得较好的识别精度。

1.2 实验要求

使用 C++ 或 Python 等语言实现模型及其训练的代码，并且不使用深度学习框架；在训练过程中使用 BP 算法和小批量训练。

1.3 实验环境

硬件环境：CPU i5-13400F，GPU：GTX 4060。

软件环境：Windows 10，Python 3.9.11，numpy 2.0.2，cupy 13.3.0。

2 分析与设计

2.1 分析思路

Mnist 数据集中的图片大小为 28×28 ，先将其铺平为长度 784 的一维向量，再输入模型中，经过三个隐藏层的学习，最后输出为 10 个数字的概率。

2.2 设计方案

全连接层模型由一层长度为 784 的输入层，三层长度分别为 256、128、64 的隐藏层和长度为 10 的输出层组成。隐藏层和输出层使用 Relu 函数进行激活。

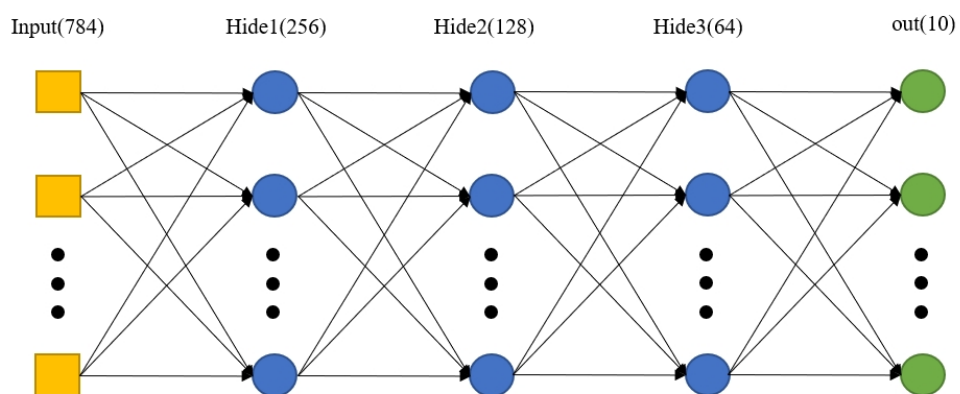


图 1: 网络架构图

3 实验实现

3.1 设计细节

3.1.1 数据集读取

读取训练集和测试集使用 `struct` 库，对于 `idx3-ubyte` 文件，先读取魔数（前四个字节）并验证，再依次读取图像数量、高度和宽度，读取后保存为 `numpy` 数据格式，并将灰度进行归一化。读取标签过程也类似。

3.1.2 模型实现

全连接层模型由一层长度为 784 的输入层，三层长度分别为 256、128、64 的隐藏层和长度为 10 的输出层组成。隐藏层和输出层使用 `Relu` 函数进行激活。初始化时，偏移值 `b` 设置为 0，权重 `W` 的元素随机初始化为 0 到 0.01 之间的数。

记 W_i 为第 i 层与第 $i+1$ 层间的权重值， b_i 为第 i 层与第 $i+1$ 层间的偏移值， A_i 为第 $i-1$ 层的输出值， Z_i 为 A_i 激活前的值，则向前传播可表示为 $Z_i = A_{i-1} \cdot W_i + b_i$ ， $A_i = \text{activate}(Z_i)$ 。

相应的，反向传播（BP）计算过程为 $dZ_i = dZ_{i+1} \cdot W_{i+1}^T \times \text{deactivate}(A_i)$ ， $dW_i = A_{i-1}^T \cdot dZ_i$ ， $db_i = \text{sum}(dZ_i, \text{axis} = 0)$ 。

3.1.3 训练函数实现

训练中，计算损失时使用交叉熵函数 $L = -\frac{1}{m} \sum_{i=1}^m \sum_{k=1}^K y_{ik} \log(\hat{y}_{ik})$ ，其中， y_k 是样本的真实标签的 one-hot 编码， \hat{y}_k 是模型预测的类别 k 的概率大小， m 是样本总数。

batch 大小为 128，epochs 为 20，学习率为 0.1。选取 batch 时，首先将所有训练样本顺序打乱，每个 epoch 按下标递增顺序抽取 128 个样本。

3.1.4 实验指标

统计模型在 `mnist` 测试集上的准确率（ACC）。

$$ACC = \frac{TP + TN}{TP + FP + FN + TN} \quad (1)$$

3.2 困难与解决方法

3.2.1 困难 1

模型收敛速度慢，经过 20 个 epochs 训练 loss 大于 1。

解决方法：将 `Sigmoid` 激活函数替换为 `Relu`。

4 实验结果与分析

4.1 实验结果

几次训练中模型的预测准确度约为 98%。最终上传的模型中，不使用 GPU 训练的模型准确度为 97.95%；使用 GPU 训练的模型准确度为 97.78%，且训练时间由 30s 左右缩短至 10s 左右。

4.2 结果分析

模型预测结果理想。