山东大学 计算机科学与技术 学院

数字图像处理 课程实验报告

|  |  |  |
| --- | --- | --- |
| 学号：201705130120 | 姓名： 苑宗鹤 | 班级： 1班 |
| 实验题目：人工智能 | | |
| 实验内容：  Cifar图像分类 | | |
| 实验过程中遇到和解决的问题：  （记录实验过程中遇到的问题，以及解决过程和实验结果。可以适当配以关键代码辅助说明，但不要大段贴代码。）  Pytorch自带的模型基本都上不去90%需要魔改输入层 或者resize到224但是会非常慢 | | |
| 实验步骤和代码：  先看一眼cuda版本    我是11.1    装pytorch  https://pytorch.org/get-started/locally/    下面按那自己的cuda版本选择  然后把command粘进去安装 我是anaconda的 python3.8.3    然后去下数据集  http://www.cs.toronto.edu/~kriz/cifar.html    下python version  image-20201123182601080  迅雷无敌了  这时候pip那边估计装好了    先来个resnet50吧    启动后会自动下载一下参数    pycharm里取消一下自动补全区分大小写 fig:      把刚才下的数据集扔到train下面  这玩意会自动帮你把cifar10的tar.gz弄好    用dataloader 把数据集load进来 batchsize看你自己的显存调 调高了炸显存    把模型转换到gpu上    使用 随机梯度下降和交叉熵损失函数    从训练集里拿到data target 就对应着网络输入 和理想结果  然后前向传播反向传播调参  直接启动 按理说可以跑了    切到任务管理器看gpu cuda      注意如果显存占用过高 或者报错了 去改batch\_size  1660ti 用256的batch\_size    然后我们建一个model用来存训练好的模型    一个来存当前的训练结果 一个来存最新的训练结果    这样我们下次加载模型 就可以从最新的参数加载了    来段可视化 可以看到如果学习率下降的不好的话会有好多时间在无效训练,  甚至过拟合    基本上输入层魔改过的网络都能上90%    测了一大堆  32x32的图片输入对于resnet这种为224\*224而造的网络来说实在是太小了  卷着卷着就没了 | | |
| 实验结果：  测试集93.5% 训练集100% | | |
| 实验总结：  1660ti跑这个实数吃力 | | |