《人工智能导论》大作业

任务名称：Mnist 条件生成器

完成组号：

小组人员：舒飞翔、麻家乐、黄天浩、张天铄、夏季

完成时间：2023.5.30

1. 任务目标

实现 Mnist 条件生成器，基于 Mnist 数据集，构建一个条件生成模型，输入的条件为 0~9 的数字，输出对应条件的生成图像。要求支持随机产生输出图像且在 cpu 上有合理的运行时间。

2. 网络模型实现细节

试验后选择分别尝试用 CGAN、GAN 和 ACGAN 进行实现，此三种模型即分别对应我组最后提交的 generate 函数中参数“pretrain”的取值“1”“2”“3”，代表选用哪种模型结构进行数字生成。

（1）CGAN：

• 生成器的网络结构

生成器使用了一个嵌入层、一个全连接层和三个反卷积层，将输入的噪声和标签扩展后通过多次转置卷积转换成 28*28 的图片。

激活函数选用 LeakyReLU，并在最后一层的反卷积层使用 tanh 以更好的解决多元生成任务。

• 判别器的网络结构

判别器同样使用了一个嵌入层、三个反卷积层和一个全连接层，嵌入层将标签向量嵌入到与图片数据相同的张量中，通过三个卷积层提取空间特征信息，在全连接层展平后进行线性变换和激活，得到输入图像是真实数据的概率。其中，卷积层进行了两次批标准化 BatchNorm2d，使数据分布更加稳定，加速收敛，一定程度上提高泛化能力。另外最后的激活函数采用的是 sigmoid，更适用于图像的二分类（真假判断）应用。

（2）GAN：

对于 GAN 模型，尝试直接部署后发现效果一般且不支持指定数字生成。因此调整思路为将 Mnist 训练集根据数字标签分为十个训练集，分别训练后获得十组生成器参数，在应用时调用对应数字的生成模型进行生成。

此方式的优势在于训练生成模型时不需要将标签作为参数，训练用的数据集更有针对性，能更好地抓住每个数字的特征。其劣势在于生成时需要加载十次模型，生成速度较慢。

训练方法与前述模型相同，区别在于需要事先对原始 Mnist 数据集进行预处理。预处理采取的方式是先将原始数据集转换为 png 格式文件再分别打包为二进制文件用于读取。过程中分别使用了 MNIST 数据集二进制格式转换为图片中转换格式部分代码以及 JPG-PNG-to-MNIST-NN-Format 项目进行格式转换。此后对各个数字分别训练即可。

（3）ACGAN：

在将 28×28 的图片矩阵传入 discriminator ，再将其与数字标签矩阵拼接生成 0、1 判别的同时，还会将图片直接传入卷积，并最后通过全连接层输出一个二十分类的概率分布，这二十类分别代表原数字类（0-9）以及生成器生成的数字类（10-19）。在计算损失函数的时候，我们希望 discriminator 可以在判断出该数字是否为生成出的数字的同时判断该数字具体是原始数字集中数据还是生成数据。即损失函数变为：

L = （L_S + L_D）*0.5

L_S = E[logP(S = real | X_{real})] + E[logP(S = fake | X_{fake})]

L_C = E[logP(C = c | X_{real})] + E[logP(C = c + 10 | X_{fake})]

其中L_s将原本训练discriminator的两次过程合并成为一次并计算均值统一优化。L_C 则是交叉熵分类损失，当传入的是 mnist 数据集中的图片时，需要判别为该类。当传入的是生成器生成的图片时，则需要判别为（该类别+10）的类。

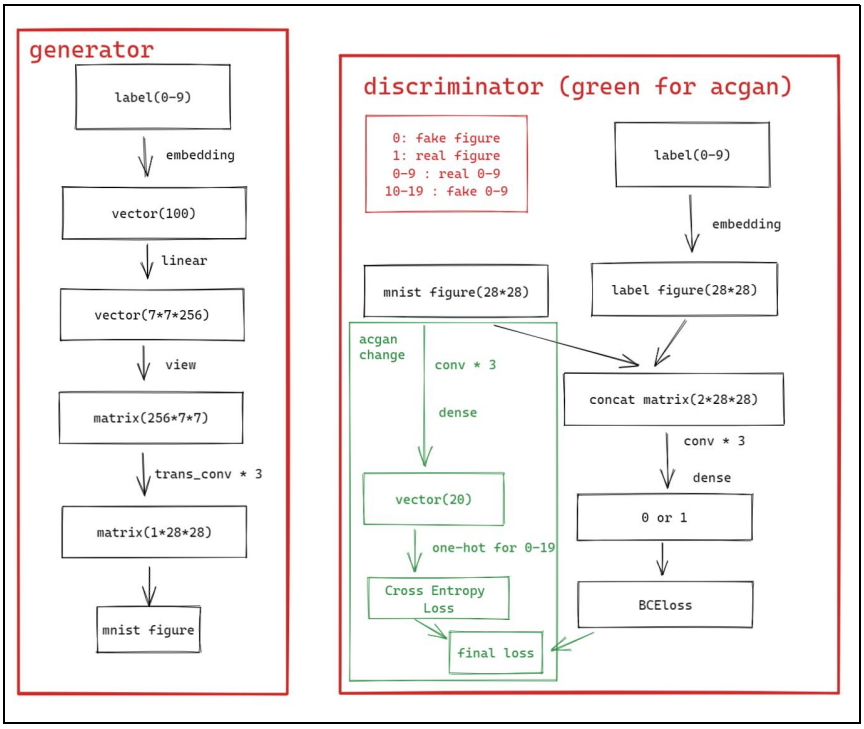

图 1　网络结构示意图

（4）分类器模型：

在经过生成模型生成数字之后，我们又训练了一个十一分类（十个数字+OOD）的 classifier 用于筛去生成样本中不好的个例。该分类器使用了三层卷积和一层全连接，最终生成一个十一个类别的概率分布向量。

训练时将原数据集随机取出十分之一样本进行相加取均值的扰动操作，形成噪声样本并标上第十一个类别的标签。通过数量足够的训练轮数之后，该分类器的识别准确率能够到达 95%以上。

3. 测试

进行可视化输出测试后可以发现：虽然大部分生成数字结果较好，但是会有少数生成效果很差。对此我们采用重训练的优化策略，设置一个准确率标准MIN_SCORE，如果生成图片经过分类器分类判定出的准确率低于这个值，就需要重新生成，通过这种方式，可以剔除掉准确率较低的生成结果。

进行部分重训练后，可以发现，部分重新训练的准确率可以保证稳定在0.8以上，剔除了准确率极低的部分。

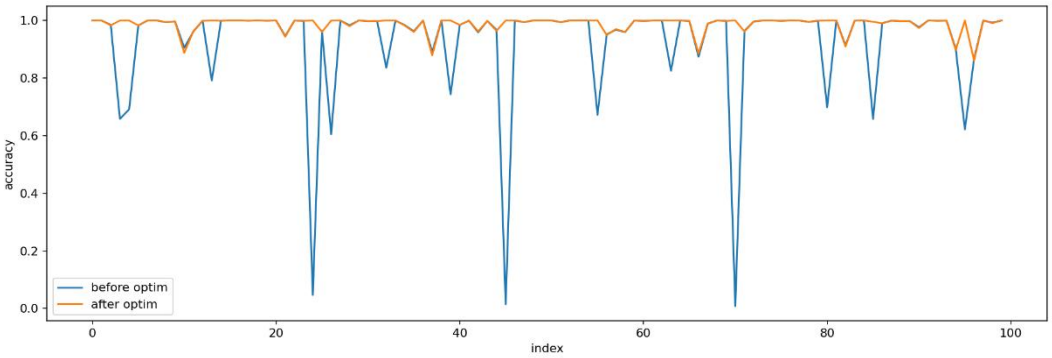

图 2　部分重训练与原始结果准确率对比

如果进行全部重训练(retrain==all)，可以达到更好的效果，

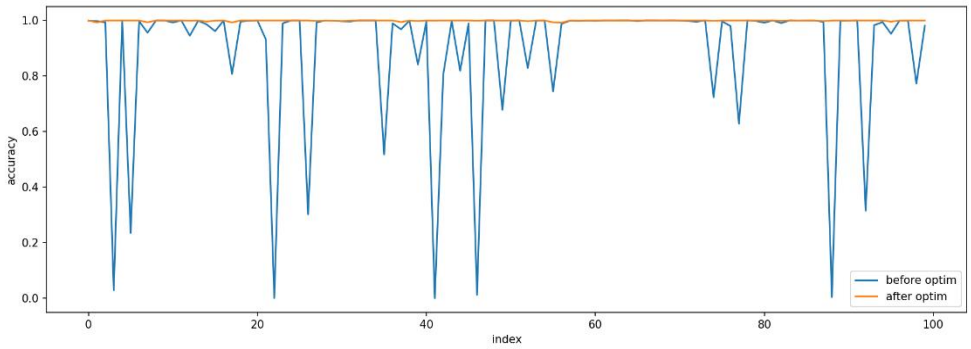

图 3 全部重训练与原始结果准确率对比

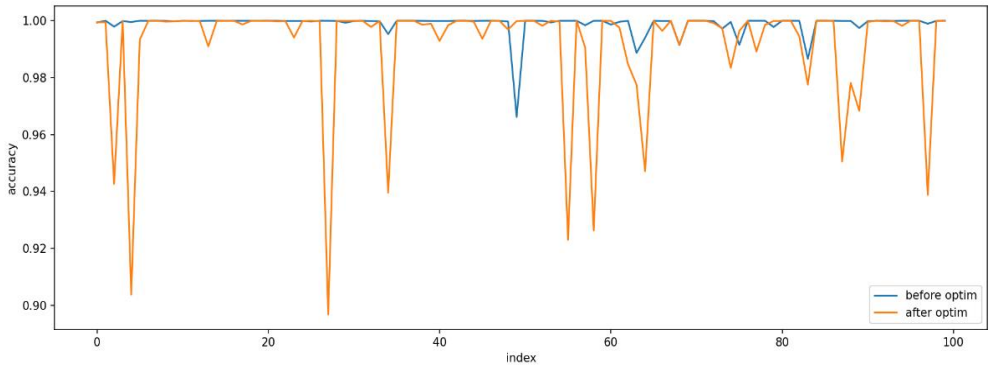

图 4 全部重训练与部分重训练准确率对比（此时纵轴为0.9-1.0）

优化轮数越多，重训练标准越高，训练结果越好，但是运行时间更长。为此，经过测试我们选择了一个较为折中的优化轮数 MAX_ITEM=10 与重训练标准 MIN_SCORE=0.85。

另外优化前后生成的数字图片从视觉来看也能发现优化后观感更好：

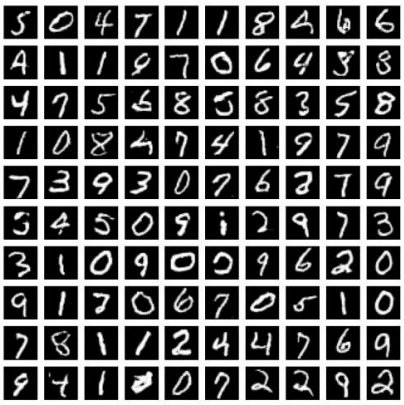

图 5 优化前生成结果

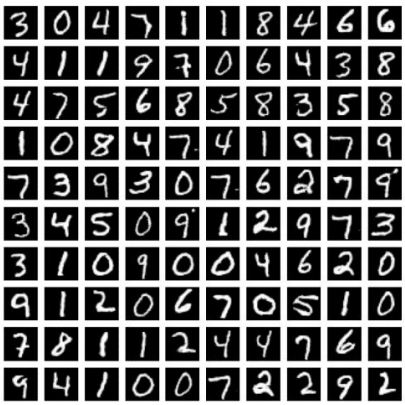

图 6 优化后生成结果

4．工作总结

（1）收获、心得

通过本次大作业，我们组研究了 GAN 模型及其衍生模型在 Mnist 数据集上的手写数字生成效果，对这类模型的网络结构、训练及测试过程、结果分析和参数调整等有了初步认识和实践经验。同时在本项目过程中，我组成员均表示感受到人工智能大有可为，并希望在后续学习中尝试将人工智能用在更复杂的任务上。

（2）遇到问题及解决思路

Label 分类讨论：

在初版接口中，当 label 为 1 时无法正常运行，检查代码发现为 squeeze 函数会将此维度压缩，故改变代码逻辑，在 generate 函数中先检查 label 长度，为 1 时不执行 squeeze，否则执行。

网络原结构激活函数效果一般：

原结构使用 ReLU 函数，相较而言 LeakyReLU 在输入为负时有一个小的负数斜率，能增加网络的非线性表示能力。另外最后一层的 tanh 激活函数能将输出限制在[-1,1]的范围内，相较于 sigmoid 函数能更好地处理中心化和标准化的数据，适合于多元分类或生成任务。

5．课程建议

在大作业之外可以考虑用实战的方式带领同学在课堂上完成一个完整任务，从而能更好地帮助同学了解项目各部分的作用并积累经验。