

**人工智能软件开发与实践**

**实验报告**

题 目 生成式对抗网络GAN

学 院 计算机科学与技术

专 业 人工智能

学 号 2021112845

学 生 张智雄

任 课 教 师 武小荷

哈尔滨工业大学计算机科学与技术学院

2023.9

**实验三:** **生成式对抗网络GAN**

1. **实验内容**

搭建Python和Pytorch环境，并在MNIST或CIFAR-10数据集上使用生成式对抗网络GAN实现图像的生成，主要包括三个部分：

* 1. 下载、预处理数据
  2. 使用Pytorch实现GAN，网络结构自行选择（DCGAN，WP-GAN等）
  3. 给出可视化生成结果

1. **算法简介及其实现细节**

## 2.1 生成式对抗网络(GAN)的基本原理

生成式对抗网络是由生成器和判别器组成的一种深度学习框架，通过对一个简单的分布采样，学习一种映射将其变换到高维复杂的训练样本分布。

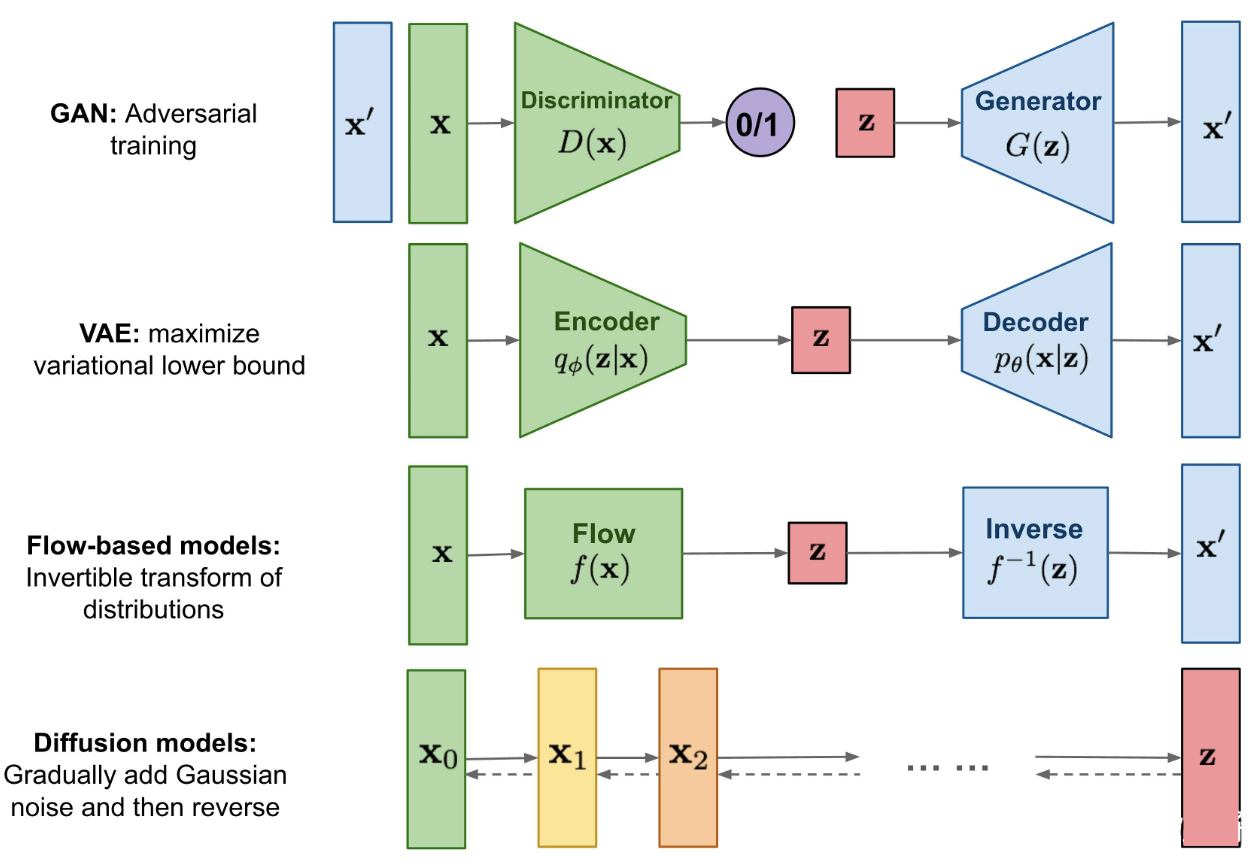


图1 生成式对抗网络模型架构

生成器通过对简单分布（如正态分布）进行采样并产生分布；然后，判别器接收生成的数据和真实数据，并尝试正确区分它们。二者通过循环迭代的对抗训练最终得到能较好拟合目标分布的生成器。

其中，对抗学习可以通过判别函数和生成函数之间目标函数的极大极小值来进行表示，一种表示方式是

其中，是从已知简单分布（如高斯分布）中的随机采样，在训练过程中，生成器和判别器交替进行更新，使得对方的误差最大化。首先固定生成器，通过最大化求解此时最优的判别器，此时和已知，则目标函数可以写成

对其求导可以解得，而后固定判别器，更新生成器以减小目标函数。代入上述最优的即可得到

可以观察得到最优情况时，当且仅当时，此时，即达到了纳什均衡。通过上述步骤交替训练生成器和判别器，GAN网络可以逐步提高合成数据的质量，从而生成更加逼真和高质量的数据样本。

但是，基于JS散度的目标函数并不能很好的表示分布之间的距离，改用不同的目标函数也能够提高逼近的效果，比如W-GAN、f-GAN等。

## 2.2 深度卷积生成对抗网络(DCGAN)的基本原理

DCGAN是GAN的一种扩展，它将CNN和GAN相结合，分别在鉴别器和生成器中显式地使用卷积层和卷积转置层。DCGAN对现有的CNN架构做了如下几个方面的修改：

1. 全卷积网络：用步幅卷积替代确定性空间池化函数，从而允许卷积提取特征的同时进行下采样。
2. 去掉全连接层，引入全局平均池化：全局平均池提高了模型的稳定性，但降低了收敛速度。
3. 批归一化： 它将每层的输入标准化为零均值和单元方差来稳定学习，这有利于由于初始化不佳产生的训练问题，有助于梯度在网络中更好的传播。

其中，判别器的输入是一个的输入图像，通过一系列Conv2d、Batch Norm2d、LeakyReLU层组成的二进制分类网络对其进行处理，最后通过Sigmoid激活函数输出最的标量概率。

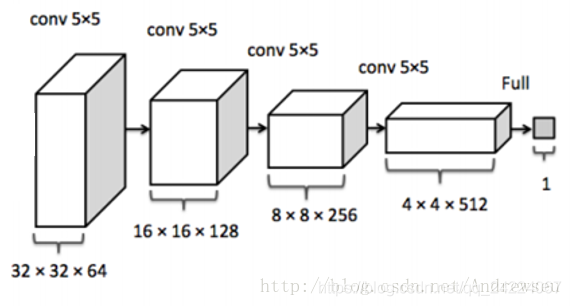


图2 DCGAN判别器结构

生成器由一系列ConvTranspose2d、BatchNorm2d层以及ReLU激活函数组成，其输入是来自标准正态分布的潜在向量，输出是大小的图像。跨行逆转置层允许将潜在向量转换成具有与图像相同形状的体积。发生器的输出通过Tanh函数馈送，使其返回到输入数据范围。

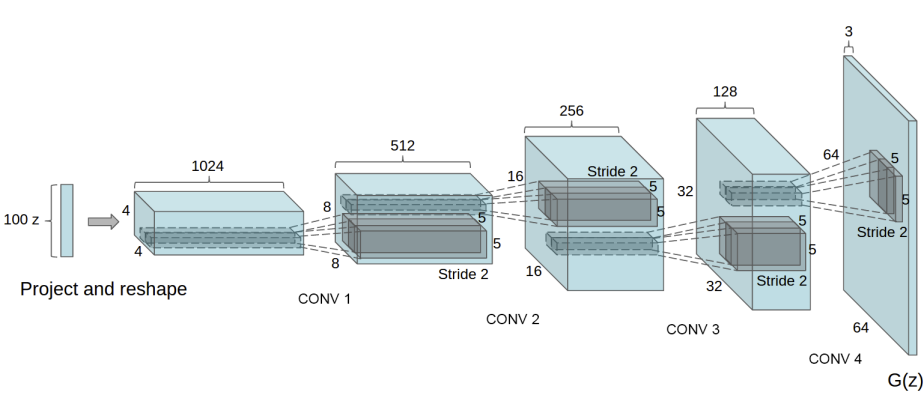


图3 DCGAN生成器结构

1. **实验设置及结果分析（包括实验数据集）**

## 3.1 实验数据集及数据预处理

在本次实验中，依旧采用实验二中介绍的MNIST和CIFAR数据集，使用torchvision自带的数据集加载MNIST和CIFAR-10数据集，并使用transforms中Resize、ToTensor、Normalize方法将图片加载为对应归一化后的Tensor张量，归一化根据数据集的通道数而定，最后通过DataLoader加载进GPU进行运算。

文本

描述已自动生成

图4 数据预处理代码

**3.2 模型设计**

在本次实验中，仿照DCGAN，实现了包含生成器和判别器的深度卷积对抗神经网络，网络的定义是重写nn.Module实现的。

3.2.1 权重初始化

在DCGAN论文中，作者指出所有模型权重应当从均值为0，标准差为0.02的正态分布中随机初始化。weights\_init函数以初始化的模型为输入，重新初始化所有卷积层、反卷积层和批标准化层，以满足这一标准。

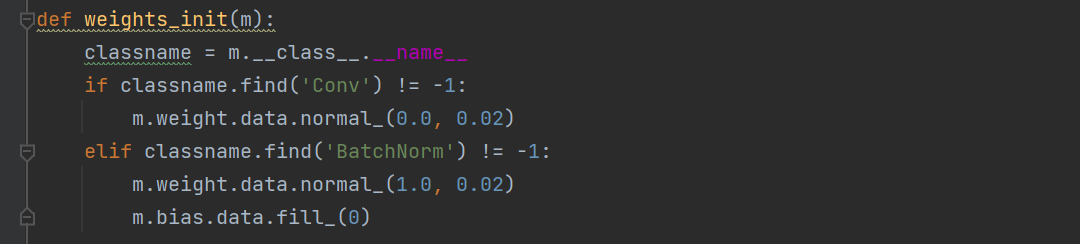


图5 权重初始化代码

3.2.2 生成器Generator

生成器通过一系列的二维反卷积层来完成的，每层都配带有批标准化层和ReLu激活。生成器的输出最终经过Tanh函数处理，以使其返回到[-1, 1]的输入数据范围。

文本

描述已自动生成

图6 生成器代码

3.2.3 生成器Discriminator

判别器是一个二分类网络，通过一系列的Conv2d，BatchNorm2d和Leaky ReLU层的处理，通过Sigmoid激活函数输出该图是真（与假相对）的标量概率。

文本

描述已自动生成

图7 判别器代码

**3.3 实验结果**

在本次实验中，在MNIST数据集上进行生成训练，使用交叉熵损失函数BCE Loss和Adam优化器，将模型输入通道根据数据集设为或者，并设置训练超参数为，为，学习率为。训练过程中判别器损失函数的值和生成器损失函数的值变化如下图8所示。

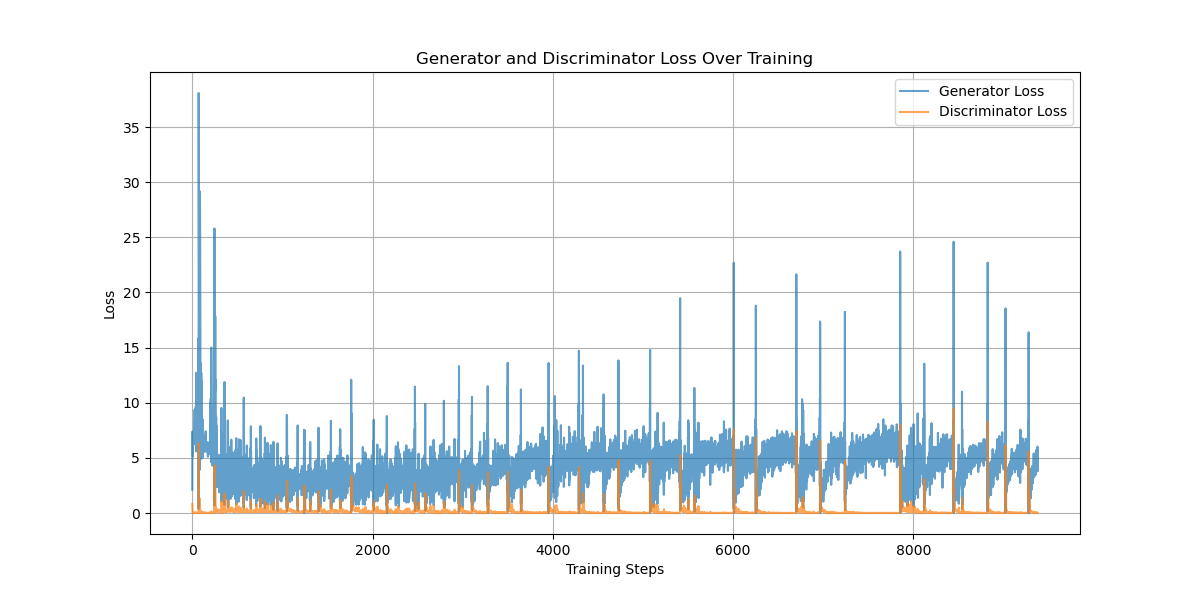


图8 损失函数变化

同时，在过程中可视化生成结果也可以发现，随训练进行，生成图像逐渐与训练集贴近，生成质量越高，但是生成图像会有略微的白点噪声，可能与模型表达能力有关。

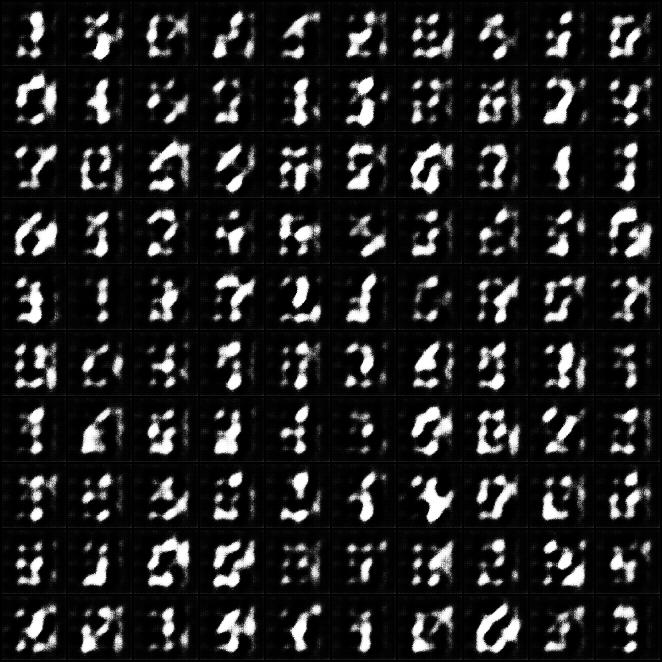


图9 训练初期生成图像



图10 训练中期生成图像



图11 训练后期生成图像

同时，在有限的时间内，还对CIFAR数据集进行了测试，但是生成效果较差，且容易出现模式坍塌或梯度消失的情况，经反复调参测试得到最好结果如下。

墙上贴着许多海报

中度可信度描述已自动生成

图12 CIFAR数据集上测试结果

分析原因可能为生成器网络在此数据集上的训练速度小于判别器，导致判别器迅速达到收敛状态，生成器训练停滞导致梯度消失。改用WGAN的损失函数后也未能得到较好的生成结果。

1. **实验收获**

本次实验中，通过实际使用调试进一步了解了深度学习框架Pytorch，在自行搭建GAN模型框架的过程中，进一步熟悉了数据加载与预处理、模型搭建模块torch.nn中Conv2d、BatchNorm2d、ConvTranspose2d等常用网络层的接口以及LeakyReLU、Tanh等函数的使用，锻炼提高了我们的代码能力。

同时，也较为深入的学习了如DCGAN、WGAN等生成对抗网络模型，了解了GAN的生成原理以及转置卷积等概念。

在代码实际编写过程中，会因为不注意维度的变化和处理、输入图像的通道数导致程序出错，同时卷积核的大小、步长以及填充也会影响训练的结果。

最后，通过实际在数据集上运行、训练GAN网络，感受到了GAN网络训练相较于其他网络较为困难，同时具有不稳定性，盲目加大训练轮数对训练结果是没有意义的（训练500个epoch，第100个epoch就梯度消失了），需要根据实际情况调整学习率、生成器和判别器的训练轮次关系和batch size（太大的batch size会导致模型学习不到较为细致的特征，从而出现模式坍塌的现象）。

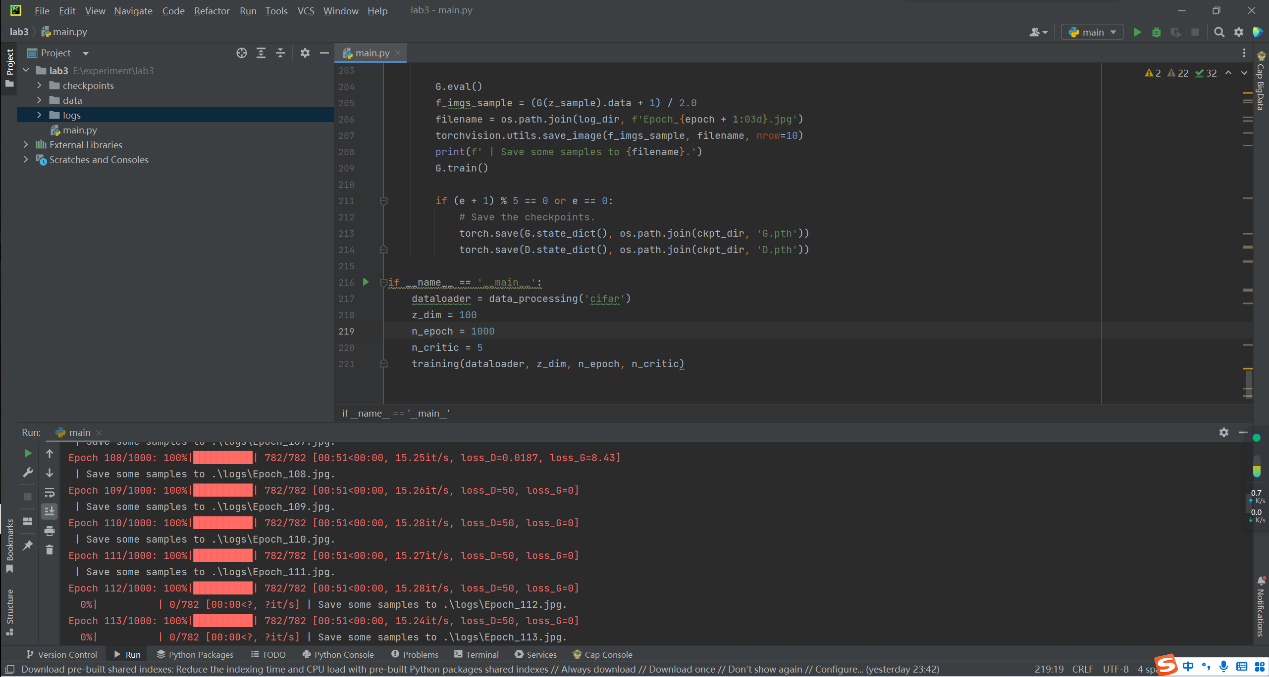


图13 梯度消失

总而言之，深度学习模型训练是一个需要耐性的过程，并且并不是随意的去调整参数就能得到较好的结果，需要根据对模型足够的了解以及对数据合适的测试有理有据的进行参数的调整。

1. **参考文献**

[1]李航. 统计学习方法[M]. 清华大学出版社, 2012.

[2]周志华. 机器学习[M]. 清华大学出版社, 2016.

[3]Goodfellow I J, Pouget-Abadie J, Mirza M, et al. Generative adversarial networks (2014)[J]. arXiv preprint arXiv: 1406.2661, 2014, 1406.

[4]Radford A, Metz L, Chintala S. Unsupervised representation learning with deep convolutional generative adversarial networks[J]. arXiv preprint arXiv:1511.06434, 2015.

[5]Arjovsky M, Chintala S, Bottou L. Wasserstein GAN[J]. arXiv preprint arXiv: 1701.07875, 2017, 685.