

DCGAN生成二次元人物头像

张卓凡 罗筱儒 丁雅杰 彭金虎

摘要

对抗生成网络(GAN)是近两年来应用于图像生成领域最多的深度模型之一。自其被提出并应用于常见图像数据集的学习生成后,出现了大量对其进行改进的变体模型。深度卷积对抗生成网络(DCGAN)模型便是利用卷积/反卷积层替换原始GAN网络得到的,我们将其应用于动漫头像图生成任务上,简单讨论了其在该任务上的适应性及改进可能性。

引言

生成模型是计算机领域的一个重要的课题。不同于分类模型等传统机器学习模型“从高维信息中得到低维信息”的特点,生成模型期望根据有限的信息生成高维信息。近些年来,生成模型应用领域最受欢迎的模型便是对抗生成网络(Generative Adversarial Network, GAN)[2]系列模型。GAN模型的思想不同于其他模型,它并不专注于直接通过对训练数据集的特征提取进行图像生成(这实际上更像是一种从高维信息到高维信息的过程),而是构造生成器(Generator)和判别器(Discriminator)两个部分。生成器以低维的高斯随机噪声向量作为输入并产生目标图像,并交由判别器进行判断。判别器比较真实数据与生成器生成的数据并进行训练,使其能正确对真实样本与生成器生成样本进行分类;固定训练后的判别器参数,训练生成器使其产生能让当前判别器分类为真实样本,之后便循环以上步骤。这个“对抗”的过程实际上是使生成样本的分布逐渐接近真实样本分布的过程。

除了最基础的GAN模型,应用于深度学习其他领域的技术方法也被引入GAN模型,其中大部分在一个或多个领域取得了不错的成果,其中的深度卷积对抗生成网络(Deep Convolutional Generative Adversarial Network, DCGAN)[1]使用反卷积/卷积层代替基础GAN模型中的全连接层,使模型具有对样

本数据的空间联系信息的提取能力,因此其在图像生成的领域的应用效果较基础GAN模型有较大的提高。本实验使用DCGAN模型进行二次元人物头像的图像生成,属于与DCGAN应用领域相贴合的应用场景。二次元人物头像具有相对复杂的信息,来源不同的图像因绘画人的画风差异,其分布又有一定的复杂性,对DCGAN的适应能力有较大的挑战,因而有实验的意义。此外,一般对生成模型的检验方法[1]是将生成模型作为数据集的信息提取器,再在生成模型的生成数据上应用线性模型,检验分类结果的准确性。但本实验的应用数据集并不具有明显的分类标签/聚类模式,因此我们仅通过可视化的方式观察模型生成样本,并根据生成样本对生成效果进行评估。

相关工作

本文主要关注GAN系列模型在图像生成领域的工作。最基础的GAN模型[1]由Goodfellow, Ian J.等人于2014年提出,并将其应用于MNIST数据集和人脸数据集的生成实验中;2015年,由Alec Radford等提出了DCGAN[2]模型,将GAN模型的全连接层替换为卷积/反卷积层,并引入Batch Normalization和LeakyReLU等技巧提高性能,应用上述模型生成MNIST数据集及LSUN卧室图片数据集,并在CIFAR-10数据集上利用线性模型验证法验证了其有效性、对卷积层进行了可视化;本文的灵感实际上来自于国立台湾大学机器学习实验室的李宏毅教授教授的“深度学习”课程的一次作业:使用GAN生成二次元人物头像。本实验结合实践情况,决定使用DCGAN代替GAN实现该任务。

数据集

本文使用的数据集源自Konachan网站的图集。在类似的课题中,有实验组从该网站中爬取数据并利用OpenCV进行头像截取,构成数据集。我们使用已有

的数据集，其共包含 50,000 张二次元人物头像图片 (96*96*3, jpg 格式)，实际作为判别器的输入时将裁剪至 (64*64*3) 减小数据量，并进行归一化处理（DCGAN-D 输入的要求，与tanh激活函数相协调）。原始数据样本如下：



Figure 1 部分数据集样本

模型方法

1. 模型概述

基于前文对GAN模型的描述，其可以规约为最小最大优化问题（假设使用交叉熵函数作为损失函数）：

$$\min_G \max_D E_{x \sim p_{data}(x)} [\log D(x)] + E_{z \sim p(z)} [1 - \log D(G(z))]$$

替换GAN模型中的全连接层为卷积/反卷积层，生成器(Generator)的网络结构如图 2 所示。

可以看到，模型以一个向量 $z \in \mathbb{R}^{100}$ 作为输入（据文献[1]，一般采用高斯随机噪声作为训练时输入），并利用多个反卷积层进行上采样操作扩增数据维度。除最后一个输出层使用双曲正切函数 \tanh 作为激励函数外，每个反卷积层使用ReLU作为激活函数，并在每次激活前进行Batch Normalization。

判别器(Discriminator)可以认为是一个基础的卷积神经网络(Convolutional Neural Network, CNN)，原文献中没有明确给出判别器的网络结果，因而我们使用与生成器反卷积层对称的卷积层（可理解为将高维信息按其产生方式逆推其低维信息），最后产生代表二分类结果的向量。除最后输出层采用Sigmoid作为激活函数外，其余各层均使用LeakyReLU作为激活函数。

2. 模型方法

Deconvolution

反卷积是上采样的主要策略，其主要作用是通过小分辨率信息产生高分辨率信息。反卷积的常见应用场合主要是在应用于图像分割的卷积神经网络中，如FCN与U-Net网络利用卷积提取特征/池化压缩特征后，需要重新获得与输入同分辨率的Heat Map，此时就会采用反卷积的策略；应用于DCGAN中的原理是相近的：我们设想一个完成训练的GAN的输入变量中每一维都代表一个特定特征的取值（如果具体到本实验，一个特征的取值可以决定动画人物脸的朝向/头发颜色等），那么反卷积层通过反卷积核采样将数据扩充到与数据集同等的维度。

Batch Normalization

Batch Normalization（以下简称BN）是由Sergey Ioffe[3]等于 2015 年提出的算法，旨在解决具有一定深度的神经网络在训练时会遇到的一个问题：Internal Covariate Shift（以下简称ICS）。关于这个问题的一个规范的定义是：在深层网络训练的过程中，由于网络中参数变化而引起内部结点数据分布发生变化。简单地说，当神经网络在训练的过程中，每次迭代更新梯度后，每一层参数的微小变化会导致下一层输入的变化，这个变化逐渐传递下去可能会导致影响被逐渐放大。这种放大后的影响可能会导致网络中使用sigmoid/tanh等饱和激活函数(saturated activation function)[4]陷入梯度饱和（即梯度逼近于 0），使得参数更新速度变得非常慢。要降低ICS的影响，一个可能的方法是去除特征之间的相关性，并将特征分布规约至均值为 0、方差为 1 的情况[3]，这也是BN算法的基本思路。通过在层与层之间增加BN处理，可以有效提高网络训练速度，并允许网络中出现饱和激励函数（DCGAN原文中，Generator中便使用了tanh作为激励函数）。

LeakyReLU

LeakyReLU是ReLU函数的变式，同样属于非饱和激励函数。相较于ReLU将所有的负值函数值均设为 0 的情况，LeakyReLU被表示为：

$$f(x) = \max(ax, x), \quad a \text{ 为可学习参数}$$

普通的ReLU可能会导致一种被称为Dead ReLU Problem的问题，即某些神经元可能永远不会被激活的问题，而LeakyReLU旨在解决这一问题。虽然从理论上推断LeakyReLU较普通的ReLU具有更好的训练效果，但并未得到大量实验结果的印证；DCGAN原文则称LeakyReLU作为Discriminator的激活函数针

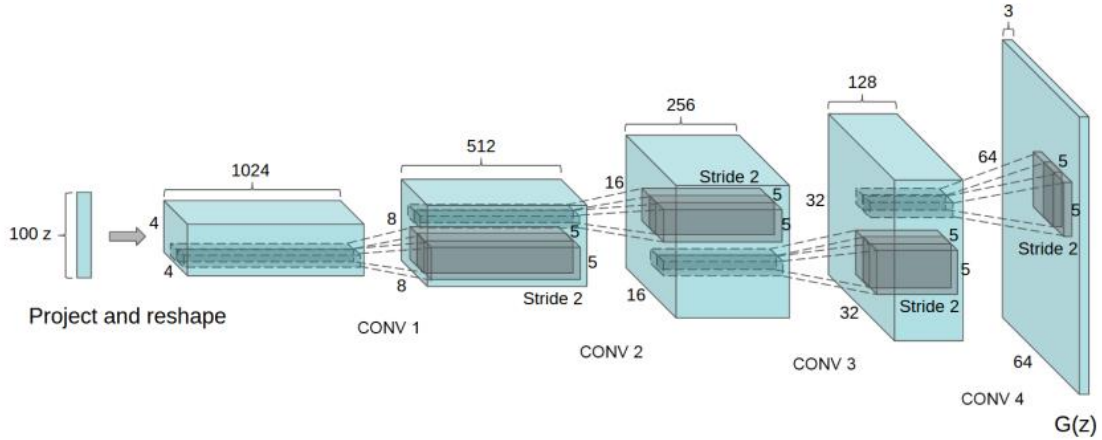


Figure 2 DCGAN-Generator结构图

对高分辨率的图像生成具有非常好的效果，结合本次实验的内容，我们决定按照DCGAN文章的提示设置LeakyReLU为激活函数。

实验流程与结果

实验使用的硬件是一台带Nvidia GeForce RTX 1070（8GB）显卡的小型工作站，使用pytorch1.0框架进行模型实现。

训练使用的损失函数为nn.BCELoss（即应用于二分类的交叉熵函数），学习算法使用Adam算法[5]，超参数的选择根据Adam算法默认推荐值设置为 $\beta = (0.5, 0.999)$ ，且Generator和Discriminator的训练学习率均设置为DCGAN文章推荐值 $l_G = l_D = 0.0002$ ，以50个样本为一个Batch，迭代训练100代（即epochs = 100）。所有可训练参数初始化为0，所有输入样本进行归一化处理。

训练的Generator产生的结果如图4所示。可以看到，在训练初期（实验中前10代训练）时，生成的样本清晰度逐渐增加、人物的五官、面部轮廓的规律逐渐被模型找到，角色脸部的用色也逐渐趋于均匀；而在中期，发色混叠不清的情况也得到改善。但从50代之后可以发现，之后每一代训练后的模型产生的样本相比于上一代的改进非常的小，并且至训练结束仍看看出其对五官正确位置确定仍存在困难，角色脸部完整的样本相对较少。我们在模型代码中加入了损失函数的可视化（如图3所示）来观察其变化，可以发现Discriminator的损失函数值每epoch有较大的抖动，且几乎没有发生下降，说明

Discriminator在对生成样本/真实样本的分类上遇到了较大的抖动问题。

反思我们遇到的问题。由于DCGAN本身引入了许多的技巧来改善模型训练困难的问题，并且由于使用了Adam学习算法（自适应学习率），已经很大程度地简化了调参过程。因此，除了模型以外，我们反思训练集的可靠性问题。由于二次元人物的形象变化较大，本质上虽然属于同一类型的数据样本，其特征却可能有非常的变异。这一问题的抽象便是训练集样本的分布是相对不集中的。回顾我们之前提到的GAN系列算法的学习逻辑，如果一个训练集有多个取值相对较高的概率密度函数区间，会导致Generator生成样本的分布很难向其中某一个/某几个集中，并有可能发生抖动。我们了解了一些普通GAN在处理该数据集时的情况，其得到的生成效果与本实验相近，并且需要接近300epoch的训练才能达到生成较好图像的目的，这从一定程度上能验证我们的猜想。之后我们又继续找寻其他数据集，但并未发现比本组使用的数据集分布更集中的训练集，故仍使用上文所提到的数据进行训练。

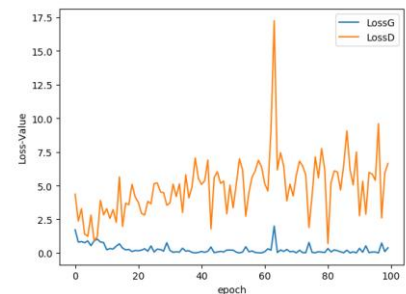


Figure 3 Loss-epoch变化图



Figure 4 生成样本例，从左到右、从上到下，epoch=1,10,30,50,75,100.

总结与讨论

本实验利用DCGAN生成二次元人物头像，经过训练后模型的Generator能产生一些具有真实绘画效果的图像，但总的生成效果不佳，可以明显看出模型未明确地拟合到一些特征。结合上文的分析，我们认为数据集的样本分布不集中可能是导致网络训练陷入停滞的一个主要原因；我们认为提高训练代数可以一定程度地带来生成样本效果的提升，但在硬件资源有限的条件下，我们没有进行进一步的尝试；我们尝试了几组超参数后效果并未比各算法及文章提供的默认超参数效果有所提升，因此我们仍只展示了默认超参数下的情况，模型结构也与DCGAN原始模型完全相同，是否可以改变各层的参数（如调节步长、增加卷积核数）来达到更好的提取参数的目的，也是值得进一步探讨的。通过本次实验我们了解了对抗生成网络的基本原理、实践了DCGAN模型的搭建，并对其针对性的优化技巧（BN、LeakyReLU）都有了初步的了解，这对于我们之后将深度学习模型应用于之后我们可能遇到的问题打下了基础。从本次实验中我们也认识到了构建优良数据集对深度学习应用的重要性，在今后的学习工作中应注意这一点。

参考文献

- [1] Radford A, Metz L, Chintala S. Unsupervised representation learning with deep convolutional generative adversarial networks[J]. arXiv preprint arXiv:1511.06434, 2015.
- [2] Goodfellow, Ian J.; Pouget-Abadie, Jean; Mirza, Mehdi; Xu, Bing; Warde-Farley, David; Ozair, Sherjil; Courville, Aaron; Bengio, Yoshua. Generative Adversarial Networks. 2014. arXiv:1406.2661
- [3] Maas A L, Hannun A Y, Ng A Y. Rectifier nonlinearities improve neural network acoustic models[C]//Proc. icml. 2013, 30(1): 3.
- [4] Goodfellow I, Bengio Y, Courville A. Deep learning[M]. MIT press, 2016.
- [5] Kingma D P, Ba J. Adam: A method for stochastic optimization[J]. arXiv preprint arXiv:1412.6980, 2014.