GAN & VQ-GAN

Tags

一.前置知识GAN

GAN的核心思想

生成对抗网络（GAN）通过**博弈论**框架实现生成模型。两个神经网络（**生成器（G）和判别器（D)**）之间对抗训练，生成器尝试生成尽可能“真实”的数据，而判别器则尝试分辨数据的真假。

最终，GAN训练效果为生成器生成的样本与真实数据分布接近，判别器无法区分数据的真假。

模型结构

生成器（G）：接受一个随机噪声 z，生成一个假数据 G(z)。

判别器（D）：接受输入数据 x（G(Z)或真实样本），并输出一个概率 D(x)，表示数据 x 来自真实数据分布的概率。

训练过程

判别器的训练：给定一个真实数据 x 和一个由生成器生成的假数据 G(z)，判别器学习判断数据是否真实。

任务是最大化真实数据的概率 D(x) 和最小化生成数据的概率 D(G(z))。

生成器的训练：生成器的目标是让判别器认为生成的数据 G(z) 是真实的。

生成器通过最小化 $log(1-D(G(z))$来使判别器对生成数据做出错误判断。

损失函数（基于二元交叉熵)

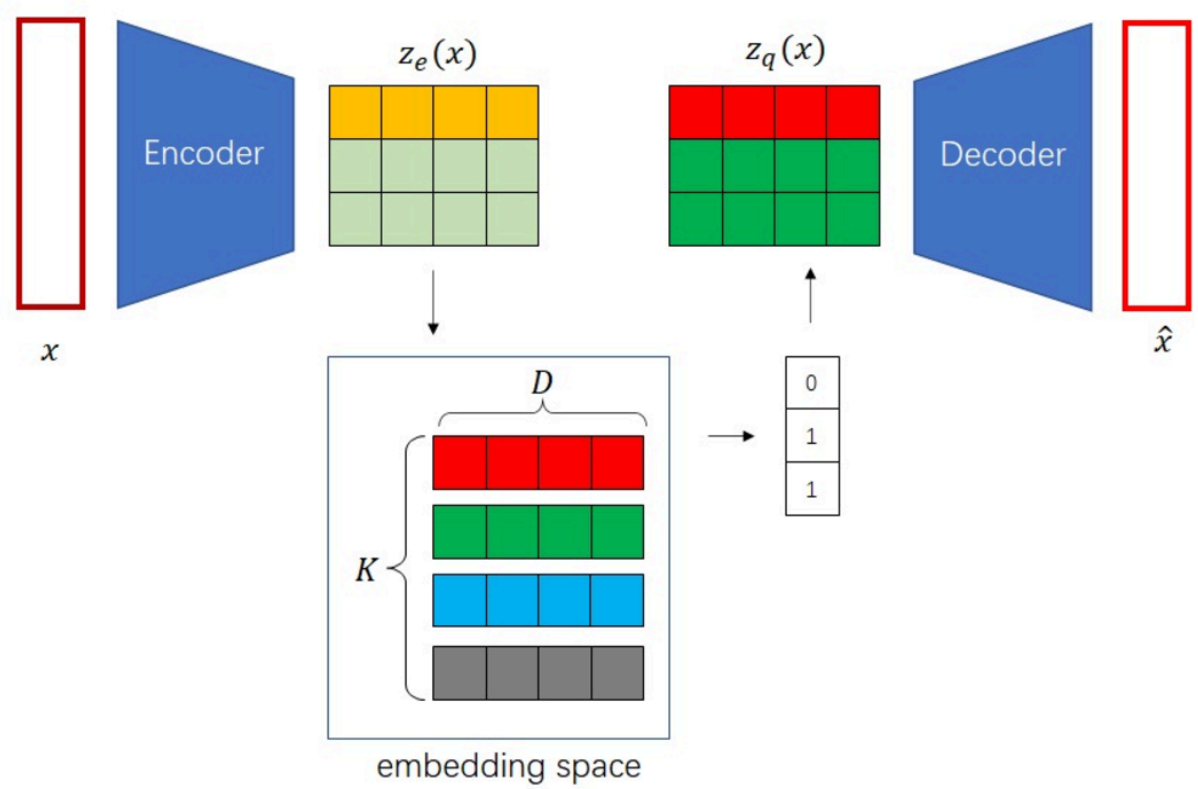
判别器的目标是最大化损失，而生成器的目标是最小化损失。整个训练过程的优化目标表示为：

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{\text{data}}(x)}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log(1-D(G(z)))]$$

二.从VQ-VAE到VQ-GAN

1.VQ-VAE

核心思想



通过引入**离散的潜在变量**（向量量化），使得潜在空间不再是连续的高维变量，而是由有限数量的离散向量（来自一个固定的代码本）组成。能够更好地捕捉和利用数据的离散结构。

损失函数

$$L = ||x - decoder(z_e(x) + sg(z_q(x) - z_e(x)))||_2^2 + \alpha||sg(z_e(x)) - z_q(x)||_2^2 + \beta||z_e(x) - sg(z_q(x))||_2^2$$

VQ-VAE的不足

VAE只使用了**均方误差**，而均方误差只能保证像素值尽可能接近，却不能保证图像的感知效果更加接近

2.VQ-GAN相比VQ-VAR的改进之处

①用感知误差(perceptual loss)代替VQ-VAE的均方误差作为VQGAN的重建误差

感知损失通过比较生成图像和真实图像在深层特征空间中的差异来度量生成图像的质量。形式为：

$$L_{\text{perceptual}} = \|\phi_i(\hat{x}) - \phi_i(x)\|_2^2$$

其中， $\hat{x}$ 是生成图像, x 是真实图像， ϕ_i 是通过预训练的神经网络提取的第 i层特征。

②引入了**GAN**的对抗训练机制，加入了一个基于图块的判别器，把**GAN**误差加入了总误差

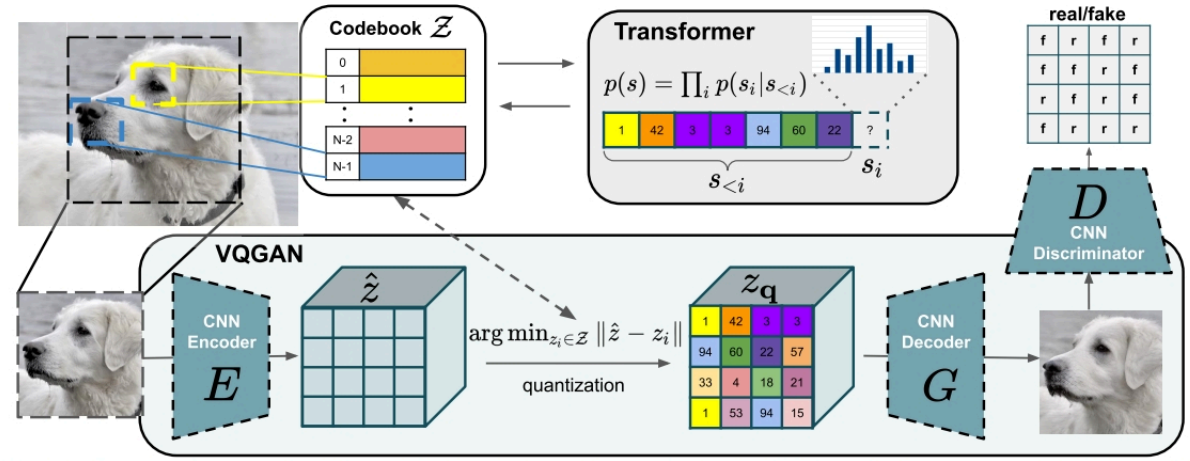
③用**transformer**代替**PixelCNN**模型作为压缩图像生成模型

三.VQ-GAN

核心思想

VQ-GAN的核心思想是通过量化技术将图像编码为离散的表示，再通过生成对抗网络（GAN）生成高质量图像。

- 训练时，先训练一个图像压缩模型（包括编码器和解码器两个子模型），再训练一个生成压缩图像的模型。
- 生成时，先用第二个模型transformer生成出一个压缩图像，再用第一个模型复原成真实图像



损失函数

$$L = L_{\text{GAN}} + L_{\text{vq}}$$

$$L_{\text{GAN}} = \mathbb{E}_{\mathbf{x}}[\log D(\mathbf{x})] + \mathbb{E}_{\hat{\mathbf{x}}}[\log(1 - D(\hat{\mathbf{x}}))]$$

注意：

$$L_{\text{vq}}^{\text{recon}} = \|\phi(\mathbf{x}) - \phi(\hat{\mathbf{x}})\|_2^2$$

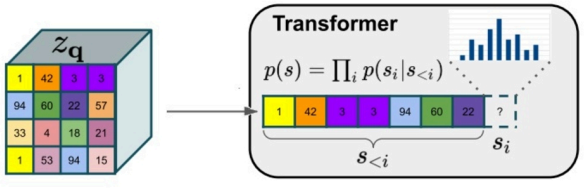
关键细节

1.基于 Transformer 的压缩图像生成模型

之前的VQVAE使用了一个能建模离散颜色的PixelCNN模型作为压缩图像生成模型。但PixelCNN的表现不够优秀。恰好，功能强大的Transformer支持建模离散的输出

问题在于怎样让像素和文字一样有**先后顺序**?

使用**自回归图像生成模型**的**常用做法**，给图像的每个像素从左到右，从上到下规定一个顺序

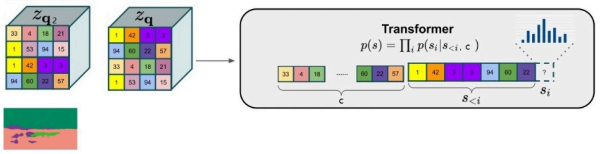


2.带约束的图像生成

①对于以图像形式表示的约束，再**训练另一个VQGAN**，把约束图像压缩成另一套压缩图片。这一套压缩图片和生成图像的压缩图片有着不同的codebook

②

给定约束的图像c，在第i步，Transformer会根据前i−1个输出像素$s_{<i}$以及c生成第i个像素 s_i。约束图像c被添加到了所有输出之前，作为这次「随机生成」的额外输入



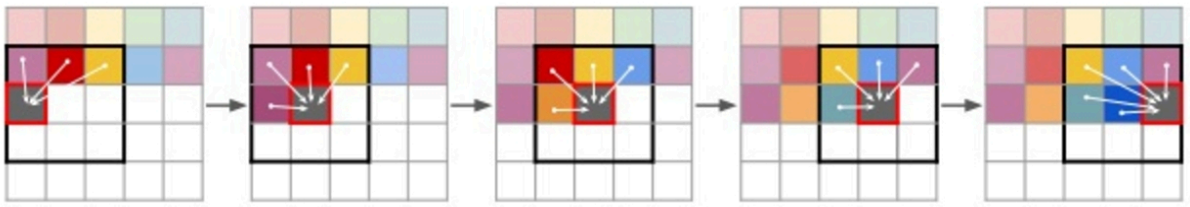
3.生成高清图像-滑动窗口机制

由于transformer注意力计算开销大，而算力资源有限，所有配置中都只使用了16×16的压缩图像。另一方面，每张图像VQGAN中的压缩比例是有限的。如果图像压缩得过多，则VQGAN的重建质量就不够好。

则该方法一次能生成的图片的最大尺寸是16f×16f，实验表明f=16的表现较好。所以该方法一次只能生成256×256的图片。这种尺寸的图片还称不上高清图片。

改进：

基于滑动窗口的采样机制来生成大图片。具体来说，把待生成图片划分成若干个16×16像素的图块，每个图块对应压缩图像的一个像素。之后，在每一轮生成时，只有待生成图块周围的16×16个图块（256×256个像素）会被输入进VQGAN和Transformer，由Transformer生成一个新的压缩图像像素，再把该压缩图像像素解码成图块。



（示意图中，每个方块是一个图块，transformer的输入是3×3个图块)