SD系列文章（一）：从AE到VAE

前沿

太久没有更新文章了，前段时间由于各种原因，文章断更了。不过影响不大，整个过程我有在自己做一些私人的笔记，这些私人的笔记后期打算通过求职面试辅导的方式开放给想从事这方面工作转行或者校招的同学，针对人群将会是对这方面算小白的同学，如果大家有想要找这方面工作的需求，可以联系我交流一下

一个很好的写文章的思路就是从提出的问题中去寻找答案，从而达到学习到知识的目的。部分知识参考别人的文章做了一下整理（我觉得写的非常好的文章，学习之后有受用）

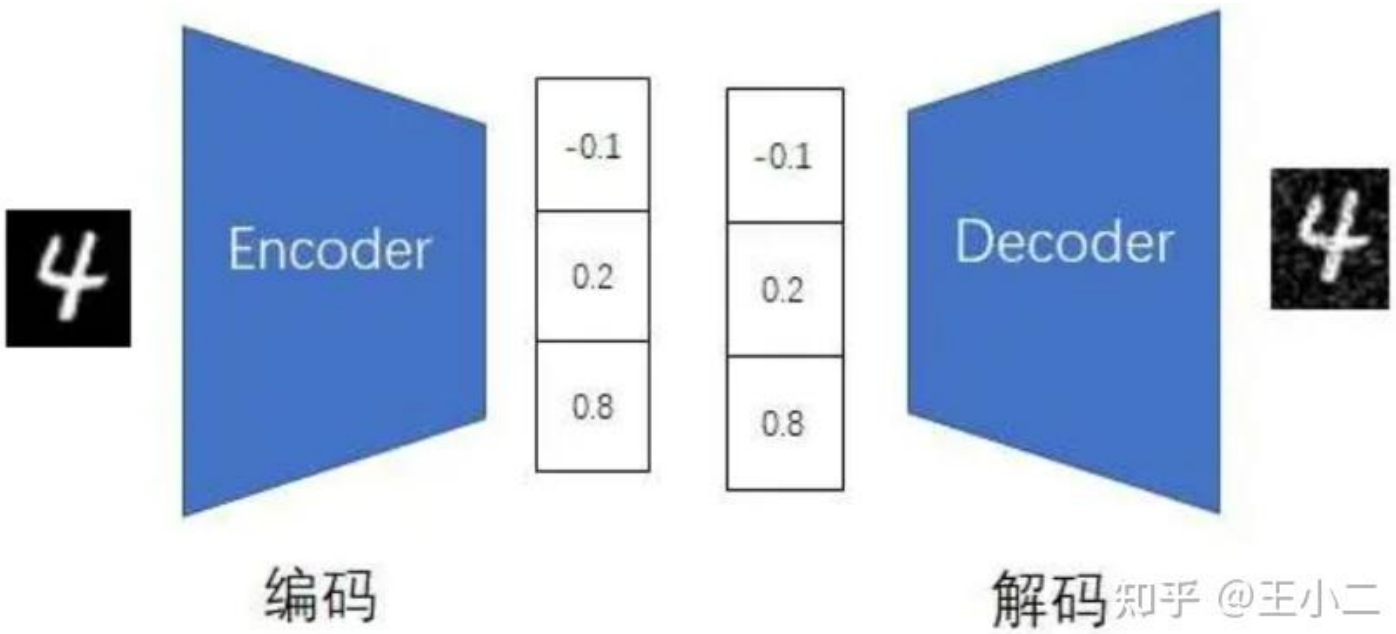
本文要解决的几个问题：

1. 什么是AE?
2. AE的作用?
3. 为什么会出现VAE？或者VAE想要解决什么问题
4. VAE为什么没有办法直接拿来做图像生成?

5．VAE为什么能作为SD的编解码器?

1、什么是AE?

AE：AutoEncoder自编码器，由一个Encoder编码器和一个Decoder解码器组成。其中编码器（Encoder）将输入的图片压缩（或者也可以理解成映射，编码）成数据，解码器（Decoder）将压缩的数据解压（或者也可以理解成重新反映射，解码）图片。下图展示了AE的一个执行流程。

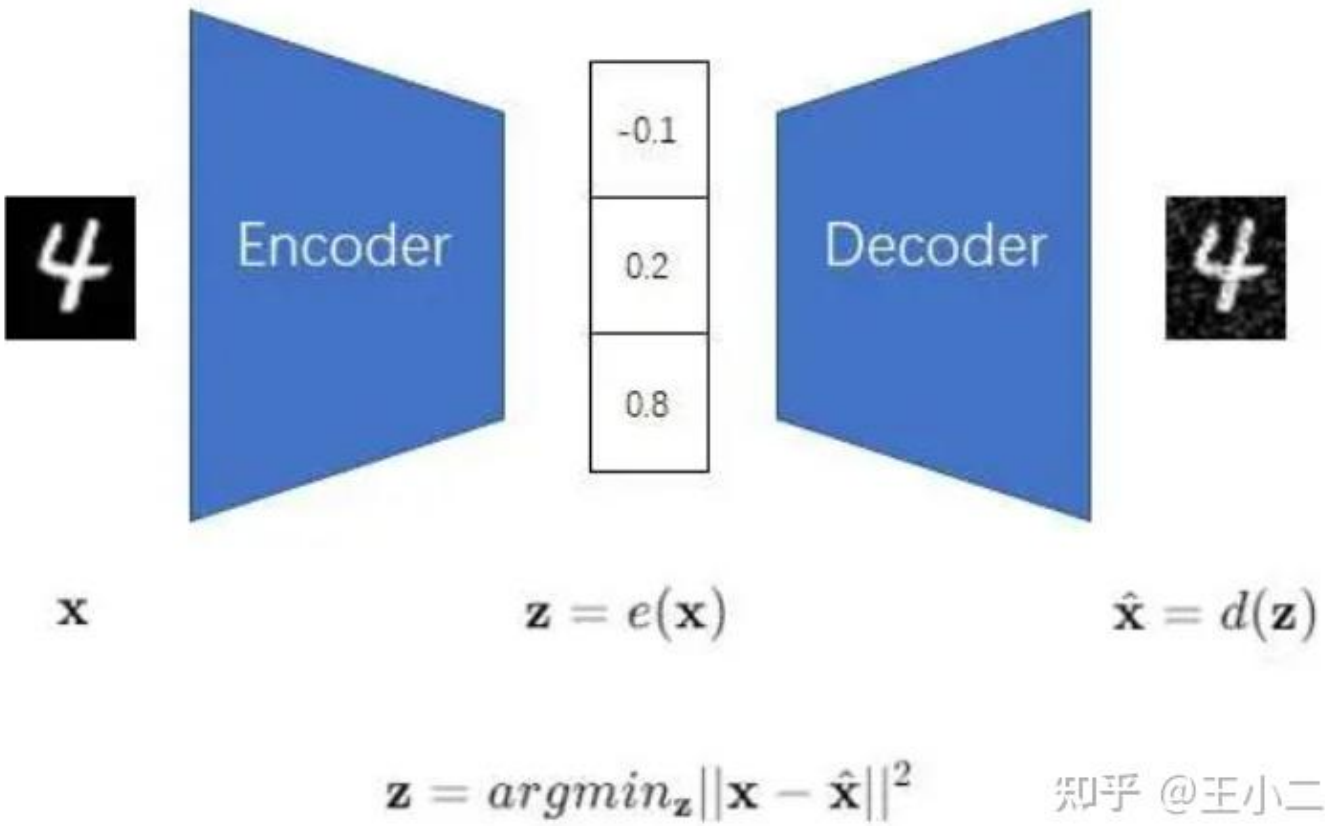


2、AE的作用?

前面我们简单了解了AE，那么AE有什么作用呢?

2.1、图像重建

实现图像重建首先最重要的点就是训练AE网络。对于输入的图像，AE的目的就是生成与原始输入数据尽可能相似的图片。因此在网络的训练过程我们的**损失函数就是计算Decoder输出的图片与输入图片的相似度**，这里可以使用均方误差来进行计算。



除了上面讲的作用之外，当然还有数据压缩等作用。

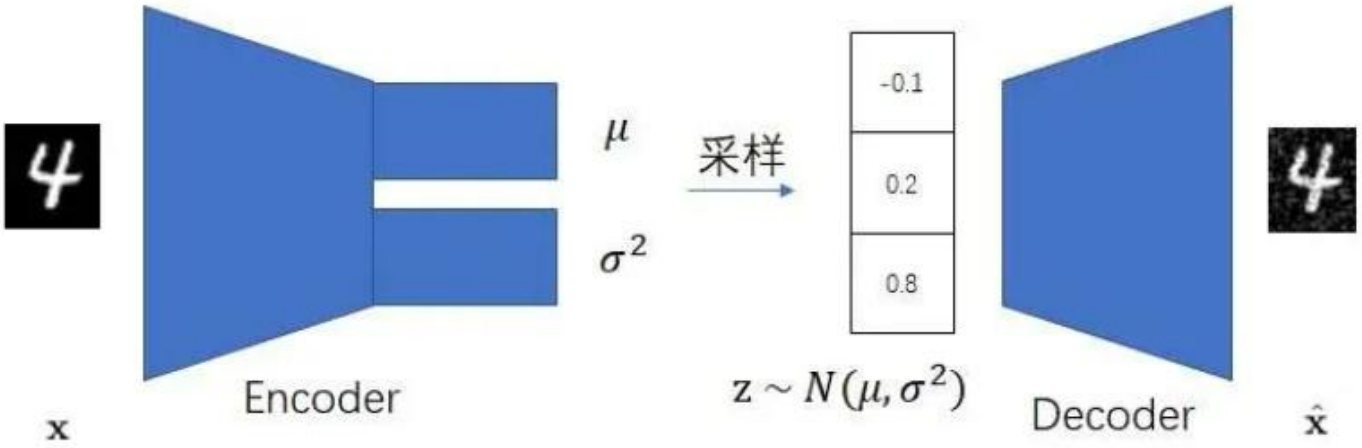
3、为什么会出现VAE？或者VAE想要解决什么问题

VAE解决AE过拟合的问题。

AE并不是一个合格的图像生成模型。我们常说的图像生成，具体是指让程序生成各种各样的图片。为了让程序生成不同的图片，我们一般是让程序根据随机数（或是随机向量）来生成图片。而普通的AE会有过拟合现象，**这导致AE的解码器只认得训练集里的图片经编码器解码出来的压缩数据，而不认得随机生成的压缩数据，进而也无法达到图像生成的要求。**

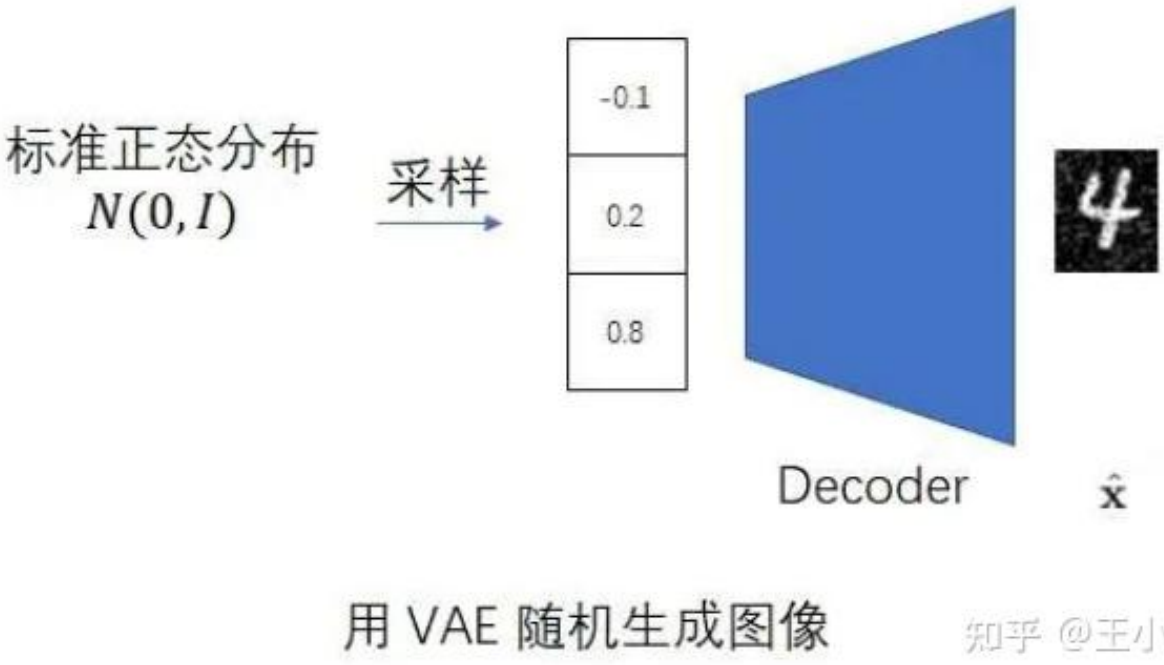
3.1、VAE的改进

第一，VAE让编码器的输出不再是一个确定的数据，而是一个正态分布中的一个随机数据。 更具体一点，训练时，编码器会同时输出一个**均值和方差** 。随后，模型会从这个均值和方差表达的正态分布里随机采样一个数据，作为解码器的输入。直观上看，这一改动就是在AE的基础上，让编码器多输出了一个方差，使得原AE编码器的输出发生了一点随机扰动。这样一来就很好解决AE对于训练数据集过拟合的问题，即只能输出看到的数据，对于未见过的数据生成效果差。



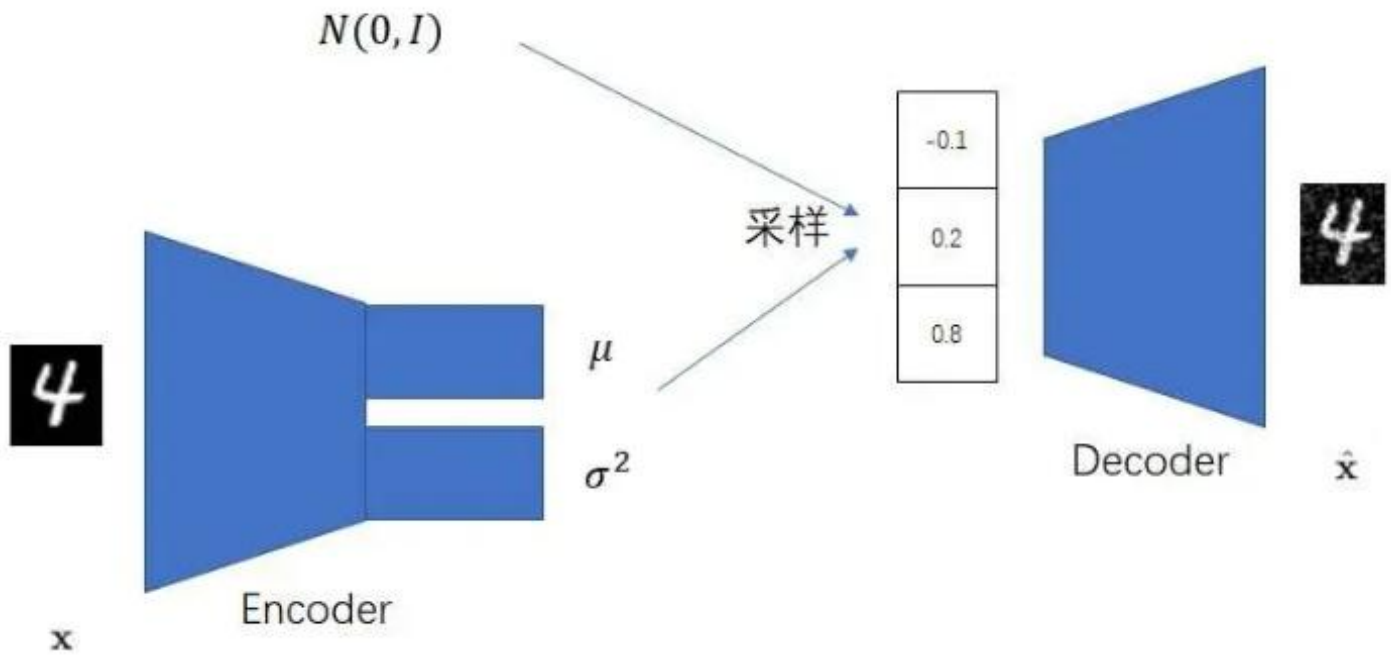
训练 VAE

VAE的第二项改动是多添加一个学习目标，让编码器的输出和标准正态分布尽可能相似。前面我们谈过，图像生成模型一般会根据一个随机向量来生成图像。最常用的产生随机向量的方法是去标准正态分布里采样。也就是说，在用VAE生成图像时，我们会抛掉编码器，用下图所示的流程来生成图像。如果我们不约束编码器的输出分布，**不让它输出一个和标准正态分布很相近的分布的话，解码器就不能很好地根据来自标准正态分布的随机向量生成图像了。**



用 VAE 随机生成图像

所以总的来说，VAE的改进有两个：1、编码器的输出一个正态分布（均值和方差），2、该分布要尽可能与标准正态分布相似。训练时VAE的Encoder部分输出均值和方差，并且从均值和方差中生成正态分布，然后从该分布中随机采样出一组数据。作为Decoder的输入。最终损失函数由Decoder输出的图片和原始输入图片的均方误差（求最小值）和Encoder生成的正态分布与标准正态分布的相似度（求最大值）组成。



loss: $||\hat{x}-x||^2 - sim(N(\mu,\sigma^2), N(0,I))$

分布与分布之间的误差可以用一个叫KL散度的指标表示。 所以，在上面那个误差函数公式中，负的相似度应该被替换成KL散度。

4、VAE为什么没有办法直接拿来做图像生成?

AE和VAE的目的都是用来重建图像，但是并没有在重建的图像的质量方面做功课。换句话说，只是让重建图像和原图像的均方误差（重建误差）尽可能小，**而没有对重建图像的质量施加更多的约束**，VAE的重建结果和图像生成结果都非常模糊。如如下图所

示，VAE的重建结果和图像生成结果都非常模糊。

5、VAE为什么能作为SD的编解码器?

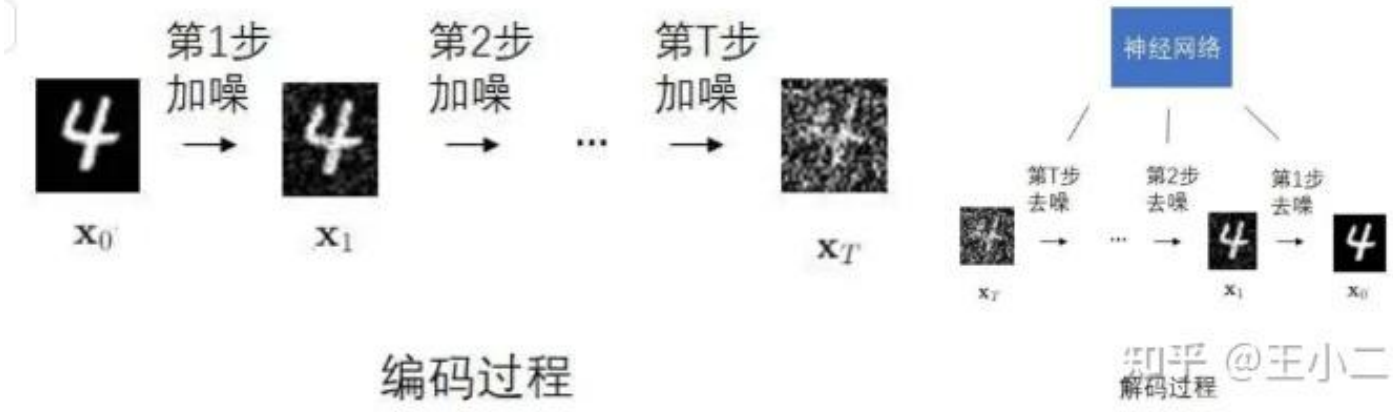
由于VAE图像生成的质量太差了，因此后续出现了DDPM这种能够用来生成高质量图像的算法。
VAE之所以效果不好，很可能是因为它的约束太少了。VAE的编码和解码都是用神经网络表示的。神经网络是一个黑盒，我们不好对神经网络的中间步骤施加约束，只好在编码器的输出（某个正态分布）和解码器的输出（重建图像）上施加约束。能不能让VAE的编码和解码过程更可控一点呢?

5.1、DDPM

DDPM的设计灵感来自热力学：**一个分布可以通过一系列简单的变化（如添加高斯噪声）逐渐变成另一个分布。** 恰好，VAE的编码器不正是想让来自训练集的图像（训练集分布）变成标准正态分布吗？既然如此，就不要用一个可学习的神经网络来表示VAE的编码器了，干脆用一些预定义好的加噪声操作来表示解码过程。

既然编码是加噪声，那解码时就应该去掉噪声。DDPM的解码器也不再是一个不可解释的神经网络，而是一个能预测若干个去噪结果的神经网络。

下图左展示了加噪的过程，右展示了去噪的过程。



1.

2.

3.

相比于VAE，DDPM的编码过程和解码过程的定义更加明确，可以施加的约束更多。因此，如下图所示，它的生成效果会比VAE好很多。同时，DDPM和VAE类似，它在编码时会从分布里采样，而不是只输出一个固定值，不会出现AE的过拟合问题。

5.2、DDPM的缺点

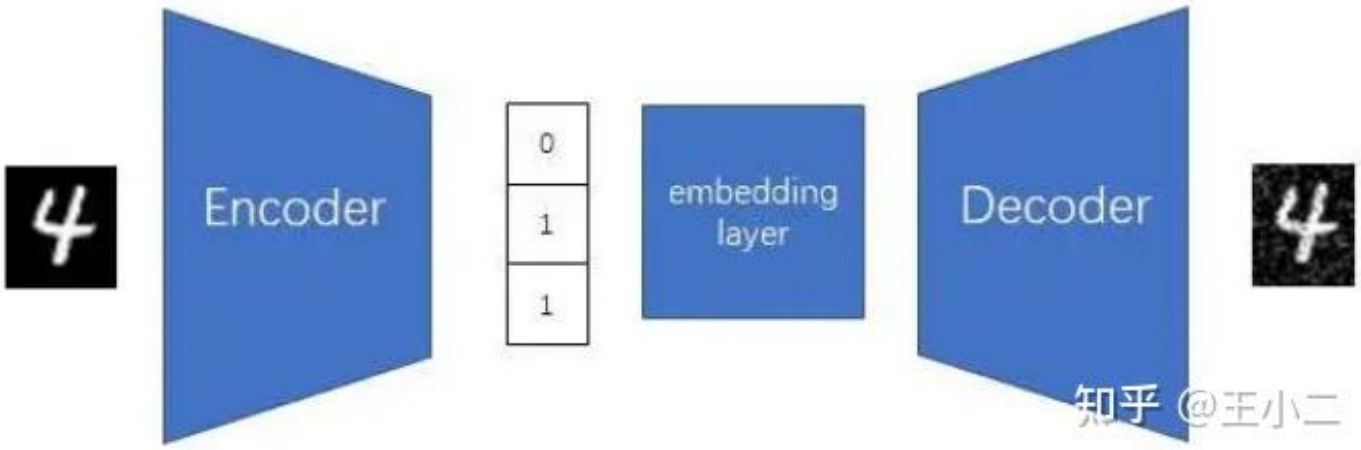
DDPM的生成效果确实很好。但是，。因此，想要用DDPM生成高质量图像，还得经过另一条路线。

5.3、VQVAE

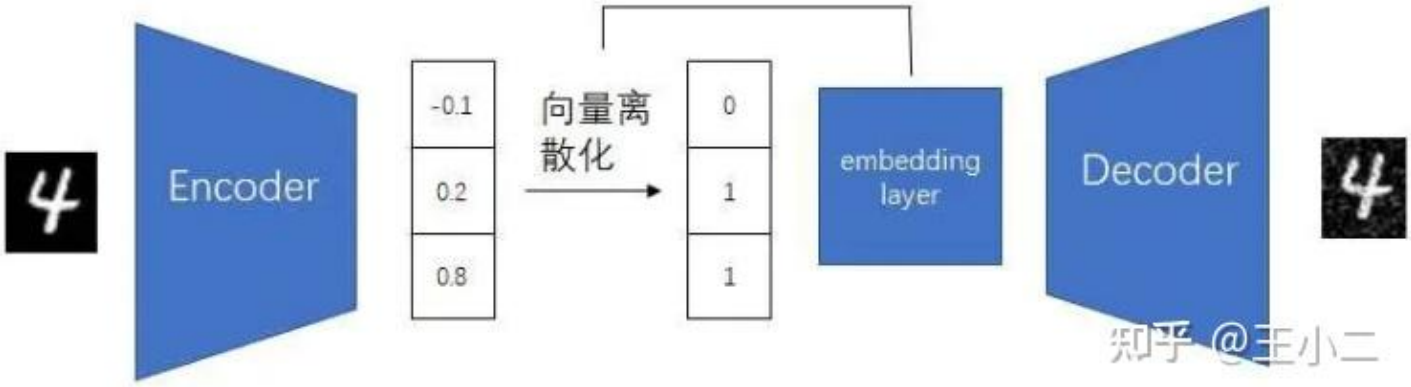
利用AE将输入的图像压缩成一个离散的向量。然后再利用Transformer等工具在压缩的空间中去生成压缩图像，最终在利用AE的Decoder将生成的图像解压缩到更大的图像。

5.3.1、Encoder输出向量的离散化

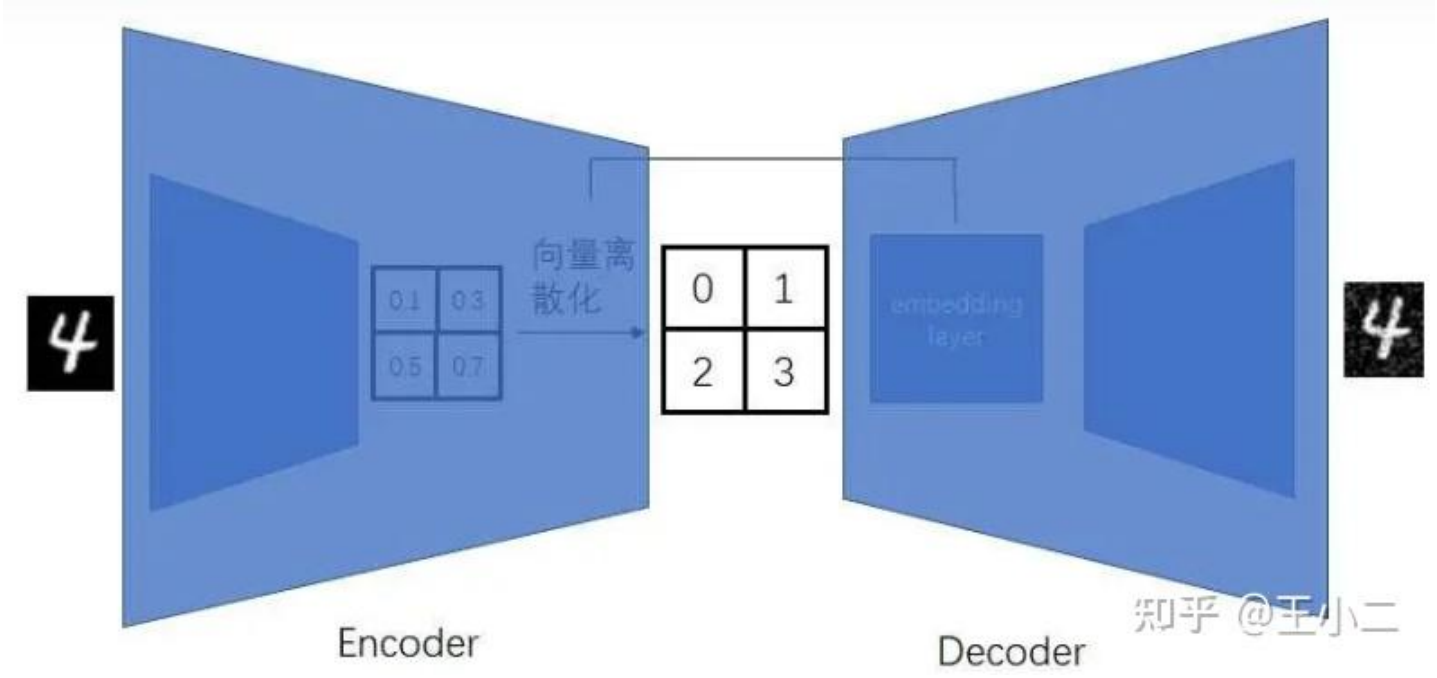
AE的编码器输出的是一个连续的向量，如何让输出的特征变为离散的呢？VQVAE的做法**就是在AE编码器的输出后面加一个把连续向量映射离散向量的嵌入层。该方法叫做离散化操作。**



整体的操作并没有修改Encoder，只不过对于Encoder的整体输出对齐到嵌入层的向量上。



但是要在压缩图像上对该压缩图像进行处理，因此我们的AE的Encoder的输出需要是一个二维的向量，能够保留原图像的一些空间特性，然后再进行二维向量离散化操作。如下图。



整理一下，VQVAE是一个能把图像压缩成离散小图像的AE。为了用VQVAE生成图像，需要执行一个两阶段的图像生成流程：

- 训练时，先训练一个图像压缩模型（VQVAE），再训练一个生成压缩图像的模型（比如Transformer）
- 生成时，先用第二个模型生成出一个压缩图像，再用第一个模型的解码器把压缩图像复原成真实图像

之所以要执行两阶段的图像生成流程，而不是只用第二个模型生成大图像，有两个原因。第一个原因是前面提到的，Transformer等生成模型只支持生成离散图像，需要用另一个模型把连续的颜色值变成离散值以兼容这些模型。第二个原因是为了减少模型的运算量。

VQVAE给后续工作带来了三条启发：第一，可以用AE把图像压缩成离散向量；第二，如果一个图像生成模型生成高分辨率的图像

的计算代价太高，可以先用AE把图像压缩，再生成压缩图像。这两条启发对应上一段提到的使用VQVAE的两条动机。
而第三条启发就比较有意思了。在讨论VQVAE的过程中，我们完全没有考虑过拟合的事。这是因为经过了向量离散化操作后，解码器的输入已经不再是编码器的输出，而是嵌入层里的向量了。这种做法杜绝了AE的死记硬背，缓解了过拟合现象。

5.4、Stable Diffusion

LDM其实就是在VQGAN方法的基础上，把图像生成模型从Transformer换成了DDPM 。或者从另一个角度说，为了让DDPM生成高分辨率图像，LDM利用了VQVAE的第二条启发：先用AE把图像压缩，再用DDPM生成压缩图像。LDM的AE一般是把图像边长压缩8倍，DDPM生成64×64的压缩图像，整套LDM能生成512×512的图像。
和Transformer不同， 。因此，在LDM中使用VQGAN做图像压缩时，不一定需要向量离散化操作，只需要在AE的基础上加一点轻微的正则化就行。作者在实现LDM时讨论了两类正则化，一类是VAE的KL正则化，一类是VQ正则化（对应VQVAE的第三条启发），两种正则化都能取得不错的效果。
LDM依然可以实现带约束的图像生成。用DDPM替换掉Transformer后，额外的约束会输入进DDPM中。作者在论文中讨论了几种把约束输入进DDPM的方式。

总结：

Stable Diffusion由两类AE的变种发展而来，**一类是有强大生成能力却需要耗费大量运算资源的DDPM**，**一类是能够以较高保真度压缩图像的VQVAE。** Stable Diffusion是一个两阶段的图像生成模型，它先用一个使用KL正则化或VQ正则化的VQGAN来实现图像压缩，再用DDPM生成压缩图像。可以把额外的约束（如文字）输入进DDPM以实现带约束图像生成。

参考：

周弈帆：Stable Diffusion 解读（一）：回顾早期工作