Diffusion Models

Plote

Diffusion Models is a probabilistic generative model based on gradually corrupting data into noise and generating data from noise by learning the inverse denoising process. It is widely used for high-quality sample generation and is trained by optimizing the variational lower bound.

September 25, 2024

Contents

1. INTRODUCTION

扩散模型已经成为最先进的深度生成模型家族。它们打破了生成对抗网络(GANs)在具有挑战性的图像合成任务中的长期统治地位，并且在各种领域也显示出潜力，包括计算机视觉，自然语言处理，时间数据建模，多模态建模，鲁棒机器学习，到计算化学和医学图像重建等领域的跨学科应用。

2. FOUNDATIONS OF DIFFUSION MODELS

Diffusion models are a family of probabilistic generative models that progressively destruct data by injecting noise, then learn to reverse this process for sample generation.

目前对扩散模型的研究主要基于三种主要的公式:

- **denoising diffusion probabilistic models (DDPMs)**
- **score-based generative models (SGMs)**
- **stochastic differential equations(Score SDEs)**

我们将对这三个公式进行独立的介绍，同时讨论它们之间的联系。

2.1. Denoising Diffusion Probabilistic Models (DDPMs)

2.1.1. Forward Diffusion Process:

A denoising diffusion probabilistic model (DDPM) makes use of two Markov chains: a forward chain that perturbs data to noise, and a reverse chain that converts noise back to data.

正向扩散过程实际上是在原始图像上逐步添加高斯噪声。这个过程可以视为对图像进行“模糊化”，每一步都会增加一定程度的噪声，最终使得图像变得不可辨认，接近于标准高斯噪声。这个过程可以看作是一个马尔可夫链

随着时间步 t 的增加，图像中的信息逐渐被噪声覆盖，直到最终形成一个几乎完全随机的噪声图像。这个过程是逐步的，每一步都可以被看作是对图像进行微小的扰动。

在每一步t,我们用一个高斯分布 $q(x_t \mid x_{t-1})$ 来描述上一步x_{t-1}到当前步骤x_t的转换,我们最终可以得到接近高斯分布的噪声

描述了数据从$q(x_0)$开始通过一系列的$q(x_t \mid x_{t-1})$生成随机变量$x_1, ..., x_T$,目标是把x_0转换为更容易处理的分布

$$q(x_1, ..., x_T \mid x_0) = \prod_{t=1}^{T} q(x_t \mid x_{t-1}) \tag{1}$$

这是扩散过程中的高斯转移核,它定义了在每一步t,如何将x_{t-1}转换到x_t,转移核是一个高斯噪声,参数β_t控制了每一步扰动的强度

公式的意思就是说x_t是x_{t-1}的一个加权和加上了高斯噪声,$\sqrt{1-\beta_t}$决定的是在x_{t-1}保留下来的数据信息

$$q(x_t \mid x_{t-1}) = \mathcal{N}\left(x_t; \sqrt{1-\beta_t} x_{t-1}, \beta_t I\right) \tag{2}$$

- 均值: $\sqrt{1-\beta_t} x_{t-1}$ 表示前一个状态 x_{t-1} 经过缩放,控制当前状态的中心位置
- 方差: $\beta_t I$ 是一个标量乘以单位矩阵I,表示添加噪声的强度,随着t增加,β_t通常增大,导致噪声逐步增大

这里的β_t在正向扩散中控制噪声强度的超参数,通常是自己设置的超参数类型:

一般有 3 种方式来设置:

- 线性调度:可以在一定范围内线性增加

$$\beta_t = \text{Max\_beta} \cdot \frac{t}{T} \tag{3}$$

  这里的"MAX_beta"是最大噪声强度,T 是总的时间步数

- 余弦调度：使用余弦函数来调整噪声强度。这种方式可以提供更平滑的变化：

$$\beta_t = \frac{1 - \cos\left(\frac{t}{T} \cdot \pi\right)}{2} \tag{4}$$

逐步将数据添加噪声，相当于逐步丢失信息，最终得到一个噪声状态。而模型也通过多次小幅添加噪声，更细致地学习每一步的变化

然后从初始状态x_0推导出任意时刻t的状态x_t,引入累计噪声参数$\overline{\alpha}_t = \prod_{s=0}^{t} \alpha_s$,我们可以从$x_0$一步生成$x_t$,不必逐步从$x_{t-1}$开始转移,可以一次性采样结果,而不需要通过完整的马尔可夫链

$$q(x_t \mid x_0) = \mathcal{N}\left(x_t; \sqrt{\overline{\alpha}_t} x_0, (1 - \overline{\alpha}_t)\mathbf{I}\right) \tag{5}$$

这里的x_t是

$$x_t = \sqrt{\overline{\alpha}_t} x_0 + \sqrt{1 - \overline{\alpha}_t} \epsilon \tag{6}$$

初始数据:x_0 高斯噪声:$\epsilon \sim \mathcal{N}(0, \mathbf{I})$

初始数据保留数据:$\sqrt{\overline{\alpha}_t}$ 噪声的强度:$\sqrt{1 - \overline{\alpha}_t}$

当扩散模型的前向过程接近结束时$\overline{\alpha}_T \approx 0$ x_T会越大接近标准高斯分布

$$q(x_T) := \int q(x_T \mid x_0) q(x_0) dx_0 \tag{7}$$

($\overline{\alpha}_T$ 是前向过程中一个关键参数，用来控制噪声的加入,是所有时间步中噪声权重的累计值,越来越接近 0, 意味着数据中几乎全部变成了噪声, 原始图像中的信息几乎完全丧失。)

随着时间步 t 的增加，数据中的信号被“淹没”在越来越多的噪声中。

2.1.2. Reverse Generation Formula

反向生成可以视为对正向扩散过程的回溯,相比于"回溯",我更喜欢使用 **engraving** 这个词来理解 diffusion。

反向生成过程开始于一个随机噪声样本，通过一系列逐步的去噪操作，逐渐生成清晰的样本。这是通过学习数据的逆扩散过程来实现的。

雕塑家从一块粗糙的石头或木材开始，通过不断地去除材料（噪点）来逐渐显现出精细的形状和细节。同样，在扩散模型的逆向生成过程中，模型从一开始的纯噪声状态（类似于一块原石）开始，通过一步步去除噪声，逐渐还原出真实的图像。

u-net 用来连接生成和去噪两大过程之间的桥梁,u-net 是靠 **epsilon** 来准确的比较两大部分之间的联系性(图像中的噪声估计值)，相当于指示了当前图像中的“多余部分”。

随着时间步的推进，噪声越来越少，当噪点趋于足够小、两张图片的噪声结构非常相似时，它们的去噪结果就会非常接近。

逆向过程中的条件概率分布,模型通过θ学习其中的均值$\mu_\theta(x_t, t)$和方差$\sum_\theta(x_t, t)$来生成上一时的图像

$$p_\theta(x_{t-1} \mid x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)) \tag{8}$$

从标准高斯分布生成初始化 x_T ,作为反向生成的起点

逐步去噪音:在每一个时间步t中,使用神经网络预测前一个状态 x_{t-1}:

$$x_{t-1} = \mu_\theta(x_t, t) + \sigma_\theta(x_t, t) \cdot \epsilon \tag{9}$$

这里 ε 是从标准高斯分布中采样的噪声，用于引入随机性。

这里的 μ_θ 和 σ_θ 是通过训练神经网络学习得到的，目标是最小化重构损失，使得模型能够准确预测每一步的去噪声过程。

重复进行去噪,从 $t = T$ 到 $t = 1$,逐步生成最终样本 x_0,最终得到的是与训练数据分布相似的有效样本

逆向过程的基本原理是逐步“去噪”,但因为前向过程中噪声的加入是有随机性的,去噪的每一步也需要保持一定的随机性。这就是为什么在逆向过程中仍然会乘以噪声。虽然我们想要去除噪声，但是为了确保生成过程中的多样性和逼真度，仍然需要引入一小部分噪声，确保每一步去噪的过程是平稳且具有随机性的。

最小化 KL 散度来匹配前向和逆向马尔可夫链

$$\text{KL}\ (q(x_0, x_1, \cdots, x_T) \parallel p_\theta(x_0, x_1, \cdots, x_T)) \tag{10}$$

前向链是加入噪声生成的过程,逆向链逐步去噪生成的过程。我们希望逆向过程能够准确地重构出初始数据。

kl 展开

$$\begin{aligned} &\text{KL}\ (q(x_0, x_1, \cdots, x_T) \parallel p_\theta(x_0, x_1, \cdots, x_T)) \\ &= -\mathbb{E}_{q(x_0, x_1, \cdots, x_T)}[\log p_\theta(x_0, x_1, \cdots, x_T)] + \text{const} \\ &= \underbrace{\mathbb{E}_{q(x_0, x_1, \cdots, x_T)}\left[-\log p(x_T) - \sum_{t=1}^{T} \log \frac{p_\theta(x_{t-1} \mid x_t)}{q(x_t \mid x_{t-1})}\right]}_{:= -L_{\text{VLB}(x_0)}} + \text{const} \\ &\geq \mathbb{E}[-\log p_\theta(x_0)] + \text{const} \end{aligned} \tag{11}$$

2.2. Score-based Generative Models (SGMs)