扩散模型 Diffusion 模型

蔚全爱

2025.8.1

1 多视角

- 从数学视角来看，扩散模型可以认为是一个齐次的随机微分方程、SDE
- 从最优传输的角度来看，扩散模型可以被理解为在概率空间中通过连续变换将简单分布（如高斯）传输到复杂数据分布（如图像分布）的一个过程
- 从高斯分布到数据分布的一种映射，VAE 实现是通过这个 Encoder 和 Decoder 的映射，而 Diffusion 模型则是通过直接 Z(隐) 到 X 的映射关系。
- 扩散模型将图像生成建模为一个从图像到噪声的正向扩散过程和一个从噪声还原图像的反向去噪过程。正向过程中，模型逐步向图像添加高斯噪声，(直接调用，不需要学习) 使其最终接近标准高斯分布；反向过程则学习逐步去除噪声，恢复出原始图像。整体过程可视为近似恒等映射，若训练充分，反向过程将收敛于真实数据分布。

2 Diffusion 模型

正向加噪：

Diffusion 模型的基本思想是从一张真实图像 x_0 出发，通过逐步添加高斯噪声，最终退化为纯噪声 $x_T \sim \mathcal{N}(0, I)$。这一过程被称为正向过程（**forward process**），其构造如下：
我们定义一个马尔可夫链 $x_0 \to x_1 \to \cdots \to x_T$，每一步满足：

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

其中：

- $q(x_t \mid x_{t-1})$ 是人为设定的、用于描述加噪过程的概率分布；
- $\alpha_t \in (0, 1)$ 是控制缩放程度的参数，接近于 1，控制均值接近漂移程度较小；
- $\beta_t = 1 - \alpha_t$，控制噪声强度，通常较小；
- 整个过程的均值逐步向 0 缩放，最终趋近于标准高斯分布；

性质：正向过程收敛至标准高斯分布

由于每一步都是高斯扰动，整个过程可以被显式写成如下形式：

$$x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1 - \bar{\alpha}_t} \cdot \epsilon, \quad \epsilon \sim \mathcal{N}(0, I) \tag{2}$$

其中，累积系数 $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s$。
我们可以看到，随着时间步数 t 增大，若 $\alpha_t < 1$，则 $\bar{\alpha}_t \to 0$，因此：

$$x_t \xrightarrow{t \to T} \sqrt{1 - \bar{\alpha}_T} \cdot \epsilon \sim \mathcal{N}(0, I) \tag{3}$$

也就是说，在充分加噪的情况下，最终的图像 x_T 会收敛为一个标准正态分布。

反向去噪：

扩散模型的目标是学习从噪声 x_T 逐步生成真实图像 x_0 的过程，即近似真实的后验分布 $q(x_{t-1} \mid x_t)$。由于该过程是马尔可夫链，我们用一个高斯分布进行近似：

$$p_\theta(x_{t-1} \mid x_t) \sim \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)) \tag{4}$$

其中 $\mu_\theta, \Sigma_\theta$ 可以设为神经网络输出，也可以将协方差固定。
我们的目标是训练神经网络，使得模型生成的分布 $p_\theta(x_0)$ 尽可能接近真实数据分布 $q(x_0)$。可以通过最小化两者的 KL 散度作为优化目标：

$$D_{\mathrm{KL}}(q(x_0) \| p_\theta(x_0)) = \mathbb{E}_{q(x_0)} \left[\log \frac{q(x_0)}{p_\theta(x_0)} \right] = C - \mathbb{E}_{q(x_0)}[\log p_\theta(x_0)] \tag{5}$$

由于项 $\mathbb{E}_{q(x_0)}[\log p_\theta(x_0)]$ 无法直接计算，我们对其进行变形。利用正向扩散过程的马尔可夫性质，有：

$$-\mathbb{E}_{q(x_0)}[\log p_\theta(x_0)] = -\mathbb{E}_{q(x_0)} \left[\log \int p_\theta(x_0, x_{1:T}) \, dx_{1:T} \right] \tag{6}$$

$$= -\mathbb{E}_{q(x_0)} \left[\log \int \frac{p_\theta(x_0, x_{1:T})}{q(x_{1:T} \mid x_0)} q(x_{1:T} \mid x_0) \, dx_{1:T} \right] \tag{7}$$

利用 Jensen 不等式和对数函数的凹性，有：

$$-\mathbb{E}_{q(x_0)}[\log p_\theta(x_0)] \leq -\mathbb{E}_{q(x_0, x_{1:T})} \left[\log \frac{p_\theta(x_0, x_{1:T})}{q(x_{1:T} \mid x_0)} \right] \tag{8}$$

即：

$$\boxed{-\mathbb{E}_{q(x_0)}[\log p_\theta(x_0)] \leq D_{\mathrm{KL}}(q(x_0) \| p_\theta(x_0)) \leq \mathbb{E}_{q(x_0, x_{1:T})} \left[\log \frac{q(x_{1:T} \mid x_0)}{p_\theta(x_0, x_{1:T})} \right]} \tag{9}$$

因此，我们可以最小化右侧的变分上界作为优化目标。

真实反向过程的表达与目标函数重写

根据贝叶斯公式，我们可以将真实的反向过程 $q(x_{t-1} \mid x_t, x_0)$ 写为如下形式：

$$\begin{aligned} q(x_{t-1} \mid x_t, x_0) &= \frac{q(x_t \mid x_{t-1}) q(x_{t-1} \mid x_0)}{q(x_t \mid x_0)} \\ &\propto \exp\left(-\frac{1}{2}\left[\frac{(x_t - \sqrt{\alpha_t} x_{t-1})^2}{\beta_t} + \frac{(x_{t-1} - \sqrt{\bar{\alpha}_{t-1}} x_0)^2}{1-\bar{\alpha}_{t-1}} \right.\right. \\ &\quad \left.\left. - \frac{(x_t - \sqrt{\bar{\alpha}_t} x_0)^2}{1-\bar{\alpha}_t}\right]\right) \end{aligned} \tag{10}$$

根据该表达式可以推导出：在已知 x_t, x_0 情况下，x_{t-1} 的条件分布仍为高斯分布：

$$q(x_{t-1} \mid x_t, x_0) \sim \mathcal{N}\left(x_{t-1}; \underbrace{\frac{\sqrt{\alpha_t}}{1-\bar{\alpha}_t} x_t + \frac{\sqrt{\bar{\alpha}_{t-1}} \beta_t}{1-\bar{\alpha}_t} x_0}_{\mu_t(x_t, x_0)}, \underbrace{\frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t} \beta_t I}_{\Sigma_t}\right) \tag{11}$$

变形目标函数　我们希望将原始的训练目标：

$$\mathcal{L} = -\mathbb{E}_{q(x_0, x_{1:T})}\left[\log \frac{p_\theta(x_0, x_{1:T})}{q(x_{1:T} \mid x_0)}\right]$$

转化为逐项可计算的形式。由于两者分布均为马尔可夫形式，可写为：

$$\mathcal{L} = \mathbb{E}_q\left[\log \frac{q(x_{1:T} \mid x_0)}{p_\theta(x_0, x_{1:T})}\right] = \mathbb{E}_q\left[\log \frac{\prod_{t=1}^T q(x_t \mid x_{t-1})}{p(x_T) \prod_{t=1}^T p_\theta(x_{t-1} \mid x_t)}\right]$$

化简得：

$$\mathcal{L} = \mathbb{E}_q\left[-\log p(x_T) + \sum_{t=2}^T \log \frac{q(x_{t-1} \mid x_t, x_0)}{p_\theta(x_{t-1} \mid x_t)} + \log \frac{q(x_1 \mid x_0)}{p_\theta(x_0 \mid x_1)}\right] \tag{12}$$

$$\begin{aligned} &= \underbrace{-\mathbb{E}_q[\log p(x_T)]}_{\mathcal{L}_T} + \sum_{t=2}^T \mathbb{E}_q\left[D_{\mathrm{KL}}\left(q(x_{t-1} \mid x_t, x_0) \| p_\theta(x_{t-1} \mid x_t)\right)\right] \\ &\quad + \mathbb{E}_q\left[D_{\mathrm{KL}}\left(q(x_0 \mid x_1) \| p_\theta(x_0 \mid x_1)\right)\right] \end{aligned} \tag{13}$$

这里我们将第一个不可导的项记为 $\mathcal{L}_T$。而由于 $p_\theta(x_{t-1} \mid x_t)$ 是高斯分布，其均值可由神经网络预测。为降低优化复杂度，我们不直接预测 x_0 或 μ_t，而是将反向过程中的均值函数重写为噪声形式。
首先定义：

$$\epsilon_\theta(x_t, t) := \frac{1}{\sqrt{1-\bar{\alpha}_t}} \left(x_t - \sqrt{\bar{\alpha}_t} \mu_\theta(x_t, t) \right) \tag{14}$$

将其代入反向过程的均值公式，反解得：

$$\mu_\theta(x_t, t) = \frac{1}{\sqrt{\alpha_t}} \left(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}} \epsilon_\theta(x_t, t) \right) \tag{15}$$

这一等式表明：我们可以用一个神经网络直接预测 $\epsilon_\theta(x_t, t)$，再计算 μ_θ 用于构造反向分布。

最终优化目标

将 μ_θ 表达式带入公式 (3-4) 中的 KL 散度项（真实后验与预测分布之间的 KL），可得最终优化目标为：

$$\mathbb{E}_{x_0,t,\epsilon} \left[\frac{\beta_t^2}{2\sigma_t^2 \alpha_t (1-\bar{\alpha}_t)} \cdot \left\| \epsilon - \epsilon_\theta \left(\sqrt{\bar{\alpha}_t} x_0 + \sqrt{1-\bar{\alpha}_t} \epsilon, t \right) \right\|^2 \right] \tag{16}$$

该表达式说明：我们训练的目标是最小化网络预测噪声与实际加噪噪声之间的误差，且带有权重项。由于 $\frac{\beta_t^2}{2\sigma_t^2 \alpha_t (1-\bar{\alpha}_t)}$ 是常数，实际实现中可以忽略，简化为：

$$\mathcal{L}_{\text{MSE}} = \mathbb{E} \left[\|\epsilon - \epsilon_\theta(x_t, t)\|^2 \right]$$

工程实现说明: 这意味着我们只需构造一个神经网络，用于预测噪声 ϵ。主流实现中常采用 U-Net 架构作为 ϵ_θ，直接以加噪图像 x_t 和时间步 t 作为输入。部分最新研究表明，采用 Transformer 结构（如 U-ViT）有望带来更好的性能。

3 偏微分视角

3.1 正向扩散过程（Forward Diffusion Process）

扩散模型的训练阶段引入一个连续时间马尔可夫过程，在该过程中逐步向数据中添加噪声。其过程可用如下的随机微分方程（**SDE**）表示：

$$\mathrm{d}\boldsymbol{x}_t = f(t, \boldsymbol{x}_t)\,\mathrm{d}t + g(t)\,\mathrm{d}\boldsymbol{w}_t, \tag{17}$$

其中：在 DDPM 等模型中，常用如下简化形式的 SDE：

$$\mathrm{d}\boldsymbol{x}_t = -\frac{1}{2}\beta(t)\boldsymbol{x}_t\,\mathrm{d}t + \sqrt{\beta(t)}\,\mathrm{d}\boldsymbol{w}_t, \tag{18}$$

这是一个线性齐次的 Ornstein-Uhlenbeck 过程（又称等方差扩散过程）。

3.2 反向生成过程（Reverse SDE）

通过 score-matching 或变分推断，可以构造反向时间的 SDE 来实现生成模型，其形式为：

$$\mathrm{d}\boldsymbol{x}_t = \left[f(t, \boldsymbol{x}_t) - g(t)^2 \nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}_t)\right] \mathrm{d}t + g(t)\, \mathrm{d}\bar{\boldsymbol{w}}_t, \tag{19}$$

其中：

- $\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}_t)$ 是关于当前状态的 score function；
- $\bar{\boldsymbol{w}}_t$ 是反向时间布朗运动；
- 扩散过程的概率密度函数 $p_t(\boldsymbol{x})$ 满足以下的 Fokker-Planck 方程（又称 Kolmogorov 向前方程）：

$$\frac{\partial p_t(\boldsymbol{x})}{\partial t} = -\nabla \cdot (f(t, \boldsymbol{x}) p_t(\boldsymbol{x})) + \frac{1}{2} \nabla^2 \left(g(t)^2 p_t(\boldsymbol{x})\right), \tag{20}$$

该方程是一个齐次偏微分方程，描述了扩散过程中概率密度随时间的演化。