扩散模型在图像重建与生成任务中的应用

李孟霖

November 2, 2025

DDPM

Denoising Diffusion Probabilistic Models(DDPM) 通过逐步添加噪声破坏图像，再用反向过程逐步去噪重建图像。该模型结合了变分下界优化和 Langevin 动力学的去噪得分匹配，实现高质量图像生成。

DDPM

数学原理

前向过程 (Forward Process):
给定原始图像 $x_0 \sim q(x_0)$，前向过程逐步添加高斯噪声，构成一个固定的马尔可夫链：

$$q(x_{1:T}|x_0) = \prod_{t=1}^{T} q(x_t|x_{t-1}), \quad q(x_t|x_{t-1}) = \mathcal{N}(\sqrt{1-\beta_t}x_{t-1}, \beta_t I)$$

可直接从 x_0 采样出任意时刻 x_t：

$$q(x_t|x_0) = \mathcal{N}(\sqrt{\bar{\alpha}_t}x_0, (1-\bar{\alpha}_t)I)$$

其中 $\alpha_t = 1 - \beta_t$, $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s$

反向过程 (Reverse Process):
从纯噪声 $x_T \sim \mathcal{N}(0, I)$ 开始，反向过程学习一个去噪马尔可夫链：

$$p_\theta(x_{0:T}) = p(x_T)\prod_{t=1}^{T} p_\theta(x_{t-1}|x_t), \quad p_\theta(x_{t-1}|x_t) = \mathcal{N}(\mu_\theta(x_t, t), \Sigma_\theta(x_t, t))$$

DDPM 损失函数构造与简化

目标：利用变分下界（ELBO），设计可优化的训练损失函数，以学习从噪声 x_T 逐步生成图像 x_0 的反向过程。

1. **变分下界（ELBO）分解：**

$$\log p_\theta(x_0) \geq \text{ELBO} = L_T + \sum_{t=2}^{T} L_{t-1} + L_0$$

- L_T 是固定常数，可忽略 - L_{t-1} 是主要的训练目标项（KL 散度）- L_0 是重建项（有时也忽略）

2. **反向过程建模为高斯分布：**

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}(\mu_\theta(x_t, t), \sigma_t^2 I)$$

- 其中 σ_t^2 为固定超参数，不学习 - μ_θ 为模型（如 U-Net）输出

DDPM 损失函数构造与简化

3. **损失函数重写为预测后验均值：**

$$L_{t-1} = \mathbb{E}\left[\frac{1}{2\sigma_t^2}\|\mu_\theta(x_t, t) - \tilde{\mu}_t(x_t, x_0)\|^2\right] + C$$

4. **使用重参数化将问题转化为噪声预测：**

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

最终损失函数简化为：

$$\mathcal{L}_{\text{simple}} = \mathbb{E}_{x_0,\epsilon,t}\left[\|\epsilon - \epsilon_\theta(x_t, t)\|^2\right]$$

结论：模型训练目标是拟合噪声 ϵ，本质上是一个时间条件件的**去噪任务**。

DDPM

模型结构

algorithm

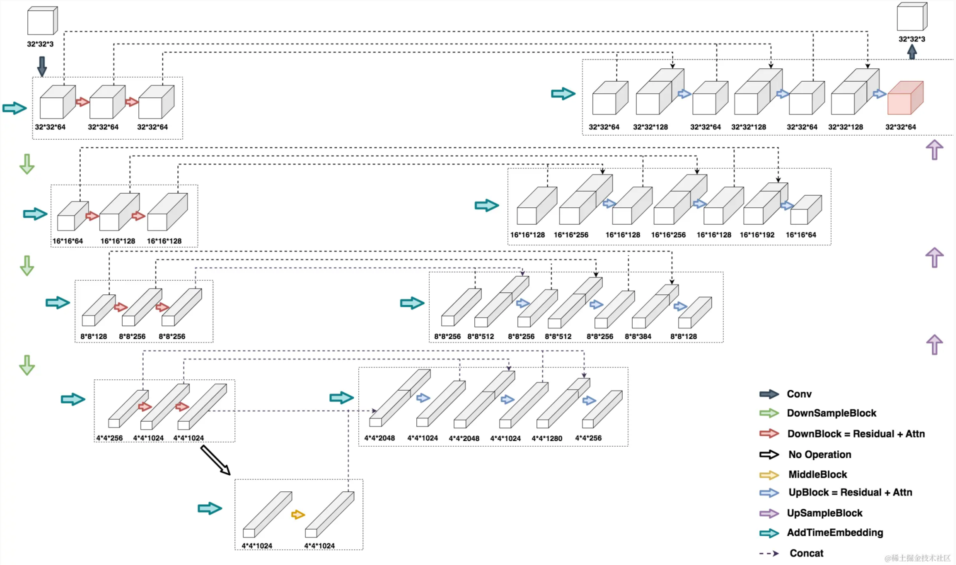
Algorithm 1 Training

repeat
  $\mathbf{x}_0 \sim q(\mathbf{x}_0)$
  $t \sim \mathrm{Uniform}(\{1, \dots, T\})$
  $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
  Take gradient descent step on
  $$\nabla_\theta \left\| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t) \right\|^2$$
until converged

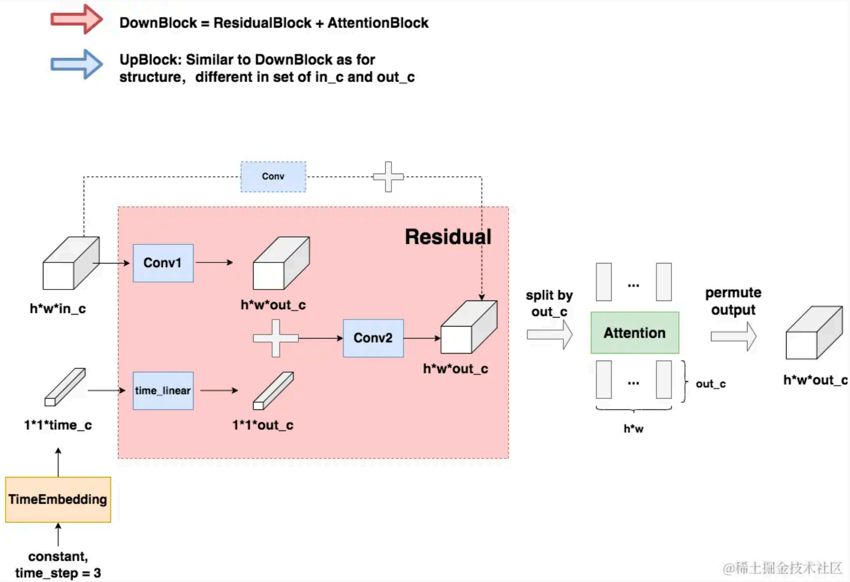
Algorithm 2 Sampling

$\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
for $t = T, \dots, 1$ **do**
  $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\mathbf{z} = \mathbf{0}$
  $\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}} \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t) \right) + \sigma_t \mathbf{z}$
return $\mathbf{x}_0$

DDPM 模型结构（U-Net）



时间步条件的注入方式


DownBlock = ResidualBlock + AttentionBlock
UpBlock: Similar to DownBlock as for structure， different in set of in_c and out_c
Conv
Residual
Conv1
h*w*in_c
h*w*out_c
Conv2
h*w*out_c
time_linear
1*1*time_c
1*1*out_c
TimeEmbedding
constant,
time_step = 3
split by
out_c
...
Attention
...
out_c
h*w
permute
output
h*w*out_c