

生成扩散模型调查

曹汉群, Cheng Tan, Zhangyang Gao, Yilun Xu, Guangyong Chen, Pheng-Ann Heng, *Senior Member, IEEE*, and Stan Z. Li, *Fellow, IEEE*

摘要 - 深度生成模型解锁了人类创造力的另一个深刻领域。通过捕获和概括数据中的模式, 我们已经进入了包罗万象的通用创造力人工智能 (AIGC) 时代。值得注意的是, 扩散模型被认为是最重要的生成模型之一, 它将人类的想法具体化为跨不同领域的有形实例, 包括图像、文本、语音、生物学和医疗保健。为了提供对扩散的先进和全面的见解, 本调查从扩散的基本公式、算法增强和扩散的多种应用三个不同角度全面阐明了其发展轨迹和未来方向。每一层都经过仔细探索, 以提供对其演变的深刻理解。这里介绍了结构化和总结的方法。

索引项 - 扩散模型、深度生成模型、扩散算法、扩散应用。

1 引言

哎呀, 我们能让机器拥有类似人类的想象力吗? 深度生成模型, 包括变分自动编码器 (VAE) [1, 2]、基于能量的模型 (EBM) [3, 4]、生成对抗网络 (GAN) [5, 6]、归一化流 (NFs) [7, 8] 和扩散模型 [9–11], 在生成真实样本方面显示出巨大的潜力。在这项调查中, 我们的中心重点在于扩散模型, 它是该领域进步前沿的缩影。这些模型有效地克服了对齐 VAE 内部后验分布所需的障碍, 减轻了 GAN 对抗目标固有的不稳定性, 解决了 EB M 训练期间与马尔可夫链蒙特卡洛 (MCMC) 方法相关的计算负担, 并执行类似于 NF 的网络约束。因此, 扩散模型在各种领域都受到了极大的关注, 包括计算机视觉 [12–14]、自然语言处理 [15, 16]、

相应的反向过程, 采用经过训练的神经网络, 通过模拟常微分方程或随机微分方程 (ODE/SDE) [11, 25] 来逐渐逆转正向过程的影响。正向过程类似于具有时变系数的简单布朗运动 [25]。神经网络经过训练, 利用去噪分数匹配目标估计分数函数 [26]。因此, 与 GAN 中采用的对抗性目标相比, 扩散模型提供了更稳定的训练目标, 并且与 VAE、EBM 和 NF 相比, 扩散模型表现出卓越的生成质量 [11, 27]。

然而, 必须承认, 与 GAN 和 VAE 相比, 扩散模型本质上需要更耗时的采样过程。这源于通过使用 ODE/SDE (普通/随机微分方程) 或马尔可夫过程将先验分布迭代转换为复杂数据分布, 因此在反向过程中需要进行大量的函数评估。此外, 其他挑战包括反向过程的不稳定性、与高维欧几里得空间训练相关的计算需求和限制, 以及似然优化所涉及的复杂性。为了应对这些挑战, 研究人员提出了不同的解决方案。例如, 已经提出了先进的 ODE/SDE 求解器来加快采样过程 [28–u201230], 同时采用了模型蒸馏策略 [14, 31] 来实现相同的目标。此外, 新颖的正向过程具有

- H. Cao is with the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong, China, also with Zhejiang Lab, Hangzhou, China. Email: 1155141481@link.cuhk.edu.hk.
- C. Tan and Z. Gao are with the AI Lab, School of Engineering, Westlake University, Hangzhou, China. Email: tancheng, gaozhangyang@westlake.edu.cn.
- Y. Xu is with Massachusetts Institute of Technology, Cambridge, Massachusetts, U.S. Email: ylxu@mit.edu.
- G. Chen is with Zhejiang Lab, Hangzhou, China. Email: gy-chen@zhejianglab.com.
- P.-A. Heng is with the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong, China.
- Stan Z. Li is with the AI Lab, School of Engineering, Westlake University, Hangzhou, China. Email: Stan.ZQ.Li@westlake.edu.cn.
- H. Cao, C. Tan, and Z. Gao contributed equally to this work.

扩散模型如何成功应用于图像合成、视频生成、3D 生成、医学分析、文本生成、语音合成、时间序列生成、分子设计和图形生成等任务。通过强调这些应用，我们旨在展示扩散模型在真实场景中的实际效用和变革潜力。

其余部分的结构如下：第 2 节概述了扩散模型的基本公式和理论。第 3 节探讨了该领域所做的算法改进，而第 4 节介绍了基于生成机制的分类应用。最后，在第 5 节中，我们总结了内容，讨论了与其他扩散调查的联系，并确定了扩散模型的局限性和未来方向。

2 个先决条件

2.1 概念和定义

2.1.1 Time and States

在扩散模型中，该过程在一条时间线上展开，该时间线可以是连续的，也可以是离散的。此时间轴中的状态表示描述模型进度的数据分布。噪声以增量方式添加到初始分布中，表示为起始状态 x_0 ，该状态从数据分布 p_0 中采样。该分布逐渐收敛于已知的噪声分布，通常为高斯分布，称为先验状态 x_T 。起始状态和先前状态之间的状态是中间状态 x_t ，每个状态都与边缘分布 p_t 相关联。这使扩散模型能够探索数据分布随时间的演变，并生成近似于先验状态 x_T 的样本。这一过程通过一系列中间状态发生，每个状态都映射到扩散过程中的特定时间点。

2.1.2 Forward / Reverse Process, and Transition Kernel

在扩散模型中，正向过程 F 将起始状态转换为先验高斯噪声，而反向过程 R 使用过渡核将先验状态去噪回起始状态。在 DDPM [10] 之后，通过定义扩散和去噪过程之间的过渡核，对扩散模型的离散公式进行了推广。

$$F(x_0, \sigma) = F_T(x_{T-1}, \sigma_T) \cdots \circ F_t(x_{t-1}, \sigma_t) \cdots \circ F_1(x_0, \sigma_1) \quad (1)$$

$$R(x_T, \sigma) = R_1(x_1, \sigma_1) \cdots \circ R_t(x_t, \sigma_t) \cdots \circ R_T(x_T, \sigma_T) \quad (2)$$

其中 F_t 和 R_t 表示时间 t 处的正向和反向转换核，噪声标度 σ 的噪声集。与归一化流动模型不同，扩散模型包含可变噪声，逐渐细化分布，以实现向目标分布的受控转变，从而提供更宽的生成空间和可控的生成。离散框架提供了连续扩散过程的离散时间近似值，允许实际实现和高效计算。

2.1.3 From discrete to continuous

当扰动核足够小时，离散过程（方程（1）和方程（2））可以推广到连续过程。[11] 表明，具有离散马尔可夫链 [9, 10] 的扩散模型可以合并到连续随机微分方程（SDE）框架中，其中生成过程反转了固定的正向扩散过程。还得出与反向 SDE 略微等效的储备 ODE [11]。连续的过程得到了更好的理论支持，并为将 ODE/SDE 社区中的现有技术应用于扩散模型打开了大门。

2.2 背景

在本小节中，我们介绍了三种基础结构，即去噪扩散概率模型、评分 SDE 公式和条件扩散概率模型，建立了与第 3 节和第 4 节的联系。以下数学公式可以被视为 2.1.2 节中一般框架的特定形式。

2.2.1 Denoised Diffusion Probabilistic Models (DDPM)

DDPM 前向处理：在 DDPM 框架中，为马尔可夫转换核选择了一系列噪声系数 $\beta_1, \beta_2, \dots, \beta_T$ ，遵循常数、线性或余弦时间表等模式，从而提高样本质量。根据 [10]，前进步骤定义为：

$$F_t(x_{t-1}, \beta_t) := q(x_t | x_{t-1}) := \mathcal{N}(x_t; \sqrt{1 - \beta_t} x_{t-1}, \beta_t \mathbf{I}) \quad (3)$$

通过从 x_0 到 x_T 的正向过渡核的组合，前向扩散过程，通过马尔可夫核 $q(x_t | x_{t-1})$ 向数据添加高斯噪声：

$$F(x_0, \{\beta_t\}_{t=1}^T) := q(x_{1:T} | x_0) := \prod_{t=1}^T q(x_t | x_{t-1}) \quad (4)$$

DDPM 逆向过程：反向过程，具有由 θ 参数化的可学习高斯核，定义为：

$$R_t(x_t, \Sigma_\theta) := p_\theta(x_{t-1} | x_t) := \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)) \quad (5)$$

μ_θ 和 Σ_θ 是反向高斯核的可学习均值和方差，由反向步长分布 p_θ 确定。从 x_T 到 x_0 的反向步骤顺序为：

$$R(x_T, \Sigma_\theta) := p_\theta(x_{0:T}) := p(x_T) \prod_{t=1}^T p_\theta(x_{t-1} | x_t) \quad (6)$$

DDPM 旨在通过联合概率分布 $p_\theta(x_0) = \int p_\theta(x_{0:T}) dx_{1:T}$ 来近似数据分布 p_0 。扩散训练目标：训练目标相当于通过引入 KL 散度 D_{KL} 来最小化负对数似然的变分界：

$$\begin{aligned} \mathbb{E}[-\log p_\theta(x_0)] &\leq \mathbb{E}_q[\underbrace{D_{KL}(q(x_T | x_0) \| p(x_T))}_{L_T} \\ &\quad + \sum_{t>1} \underbrace{D_{KL}(q(x_{t-1} | x_t, x_0) \| p_\theta(x_{t-1} | x_t))}_{L_{t-1}} \\ &\quad - \underbrace{\log p_\theta(x_0 | x_1)}_{L_0}] \end{aligned} \quad (7)$$

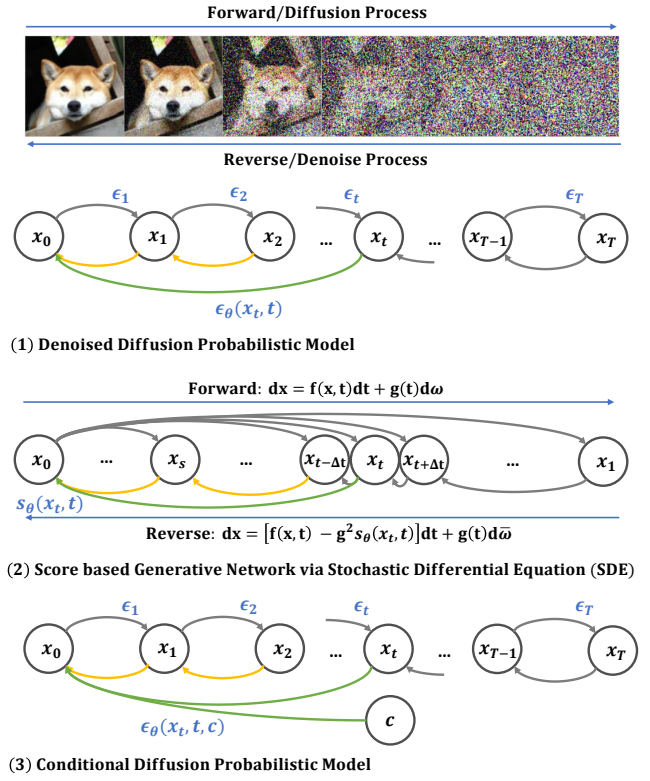
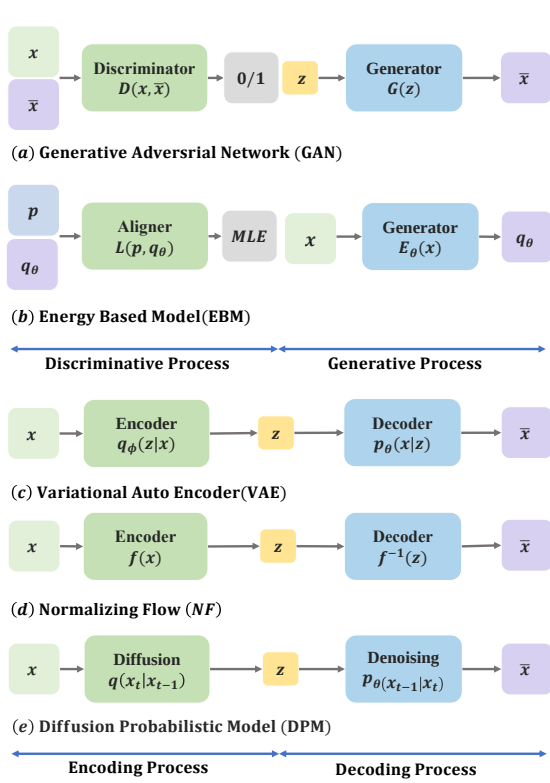


图 1. 扩散模型概述。(左) 生成模型简介。GAN 和 EBM 首先利用生成器来捕获数据分布。通过从训练数据中采样，根据实际分布对估计的分布进行校正。分布捕获过程和鉴别标准不同。VAE、NF 和 DPM 通过编码过程直接将实际分布投影到预定义分布中。实例是通过解码预定义分布中的样本来获取的。他们应用了不同的预定义分布 z 和编码和解码过程。(右) 扩散模型的简化公式。一般过程遵循右上角的数字。数据分布被扩散为随机的高斯噪声，并通过去噪进行反转。(1) DDPM (Section 2.2.1) 实现了沿离散时间线的逐步扩散和去噪过程。(2) SDE (Section 2.2.2) 建立连续时间线，通过基于函数实现国家间转换

其中 L_T 和 L_0 表示先前的损失和重建的损失; $L_{1:T-1}$ 表示同时前进和反向步骤的后验之间的散度和。简化 L_{t-1} ，我们根据后验 $q(x_{t-1}|x_t, x_0)$ 得到名为 L_{simple} 的简化训练目标：

$$q(x_{t-1}|x_t, x_0) = \mathcal{N}(x_{t-1}; \tilde{\mu}_t(x_t, x_0), \tilde{\beta}_t I) \quad (8)$$

其中 $\tilde{\beta}_t$ 取决于 β_t 。保持上述参数化并将 x_t 重新参数化为 $x_t(x_0, \epsilon)$ ， L_{t-1} ，表示为两个平均系数之间 l_2 损失的期望值：

$$L_{t-1} = \mathbb{E}_q \left[\frac{1}{2\sigma_t^2} \|\tilde{\mu}_t(x_t, x_0) - \mu_\theta(x_t, t)\|^2 \right] + C \quad (9)$$

这与下一段中讨论的降噪分数匹配相关联。通过重新参数化 μ_θ w.r.t ϵ 来简化 L_{t-1} ，简化名为 L_{simple} 的训练目标：

$$L_{simple} := \mathbb{E}_{x_0, \epsilon} \left[\frac{\beta_t^2}{2\sigma_t^2 \alpha_t (1 - \bar{\alpha}_t)} \left\| \epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t} x_0 + \sqrt{1 - \bar{\alpha}_t} \epsilon) \right\|^2 \right] \quad (10)$$

大多数扩散模型使用 DDPM 训练策略。但是，改进的 DDPM 建议将 L_{simple} 与其他目标相结合。训练后，预测网络 ϵ_θ 用于反向过程进行祖先采样。

2.2.2 Score SDE Formulation

Score SDE [11] 将 DDPM 中的离散时间方案扩展到基于随机微分方程的连续时间框架。此外，它还提出了基于 ODE 公式的附加确定性采样框架。

正向 SDE：ScoreSDE [11] 连接了连续差异融合过程和随机微分方程。相反的过程与 It 的解 $\hat{\phi}$ SDE [39] 相关联，该解由平均偏移的漂移项和用于额外噪声的布朗运动组成：

$$dx = f(x, t)dt + g(t)dw, t \in [0, T] \quad (11)$$

其中 w_t 是标准维纳过程， $f(\cdot, t)$ 是 $x(t)$ 的漂移系数， $g(\cdot)$ 是独立于 x 的简化扩散系数。 $p_t(x)$ 和 p_T 分别表示边际分布和先验分布。如果系数是分段连续的，则正向 SDE 方程具有唯一的解 [40]。提出了两种类型的正向过程：变异保留 (VP) 和变异爆炸 (VE) SDE。VP 对应于

DDPM 框架：

$$\begin{aligned} \text{VP: } dx &= -\frac{1}{2}\beta(t)x dt + \sqrt{\beta(t)}dw \\ \text{VE: } dx &= \sqrt{\frac{d[\sigma^2(t)]}{dt}} dw \end{aligned}$$

反向 SDE：扩散模型的采样是通过正向过程的相应反向时间 SDE 完成的（方程（11））[41]：

$$dx = [f(x, t) - g(t)^2 \nabla_x \log p_t(x)] d\bar{t} + g(t) d\bar{w}, t \in [0, T] \quad (12)$$

其中 w_t 是标准维纳过程， $f(\cdot, t)$ 是 $x(t)$ 的漂移系数， $g(\cdot)$ 是简化的扩散系数。 $p_t(x)$ 和 p_T 是边际分布和先验分布。如果系数是分段连续的，则正向 SDE 方程存在唯一的解。[42]：

$$L := \mathbb{E}_t \{ \lambda(t) \mathbb{E}_{x_0} \mathbb{E}_{q(x_t|x_0)} [\|s_\theta(x_t) - \nabla_{x_t} \log p(x_t|x_0)\|_2^2] \} \quad (13)$$

其中 x_0 是从分布 p_0 中采样的， $\lambda(t)$ 是正加权函数，以保持随时间变化的损失处于相同的幅度 [11]。 $q(x_t|x_0)$ 是与方程（11）中的前向过程相关的高斯瞬态核。例如， $q(x_t|x_0) = \mathcal{N}(x_t; x_0, \sigma^2(t)I)$ 。可以证明，去噪分数匹配目标（方程（13））中最优解等于几乎所有 x, t 的真实分数函数 $\nabla_x \log p_t(x)$ 。此外，评分函数 s_θ 可以看作是 DDPM 目标中神经预测 ε_θ 的重新参数化（方程（10））。[43] 进一步表明，扩散模型正向过程中的 SCORE 函数可以分解为三个阶段。当从近场移动到远场时，扰动数据会受到数据分布中更多模式的影响。

概率流 ODE：概率流 ODE [11] 支持与 SDE 共享相同边际概率密度的确定性过程。受 Maoutsa *et al.* [44] 和 Chen *et al.* [45] 的启发，任何类型的扩散过程都可以衍生成特殊形式的 ODE。方程（12）的相应概率流 ODE 为

$$dx = \{f(x, t) - \frac{1}{2}g(t)^2 \nabla_x \log p_t(x)\} dt \quad (14)$$

与 SDE 相比，概率流 ODE 可以用更大的步长来解决，因为它们没有随机性。因此，PNM [46] 和 DPM-Solver [47] 等一些工作基于高级 ODE 求解器获得了更快的采样速度。

2.3 条件扩散概率模型

扩散模型是通用的，能够从无条件 p_0 和条件 $p_0(x|c)$ 分布中生成数据样本，其中 c 作为给定条件，例如类标签或链接到数据 x [36] 的文本。分数网络 $s_\theta(x, t, c)$ 在训练期间整合了这个条件。各种采样算法，包括无分类器指导 [48] 和分类者指导 [27]，都是为条件生成而设计的。

标记条件 使用标记条件进行采样将指导每个采样步骤的梯度。它通常需要一个具有 UNet Encoder 架构的附加分类器来为特定标签生成条件梯度，这可以是

文本、分类、二进制或提取的特征 [12, 27, 28, 49–55]。该方法首先由 [27] 提出，是当前条件采样技术的基础。

未标记条件 未标记条件抽样使用自我信息作为指导，通常以自我监督的方式应用 [56, 57]。它通常用于去噪 [58]、绘画到图像 [59] 和修复任务 [17]。

3 算法改进

尽管跨不同数据模态生成了高质量的扩散模型，但它们的实际应用还有待改进。与 GAN 和 VAE 等其他生成模型不同，它们需要一个缓慢的迭代采样过程，并且它们的正向过程在高维像素空间中运行。本节重点介绍了增强扩散模型的四个最新进展：（1）Sam-pling 加速技术（第 3.1 节）以加速标准 ODE/SDE 模拟；（2）新的前向过程（第 3.2 节），用于改进像素空间中的布朗运动；（3）增强扩散 ODE 似然的似然优化技术（第 3.3 节）；（4）桥接分布技术（第 3.4 节），利用扩散模型概念来连接两个不同的分布。

3.1 采样加速

尽管扩散模型生成保真度很高，但其实际用途受到采样速度慢的限制。本节简要概述了提高采样速度的四种高级技术：蒸馏、训练计划优化、无训练加速以及扩散模型与更快的生成模型的集成。

3.1.1 Knowledge Distillation

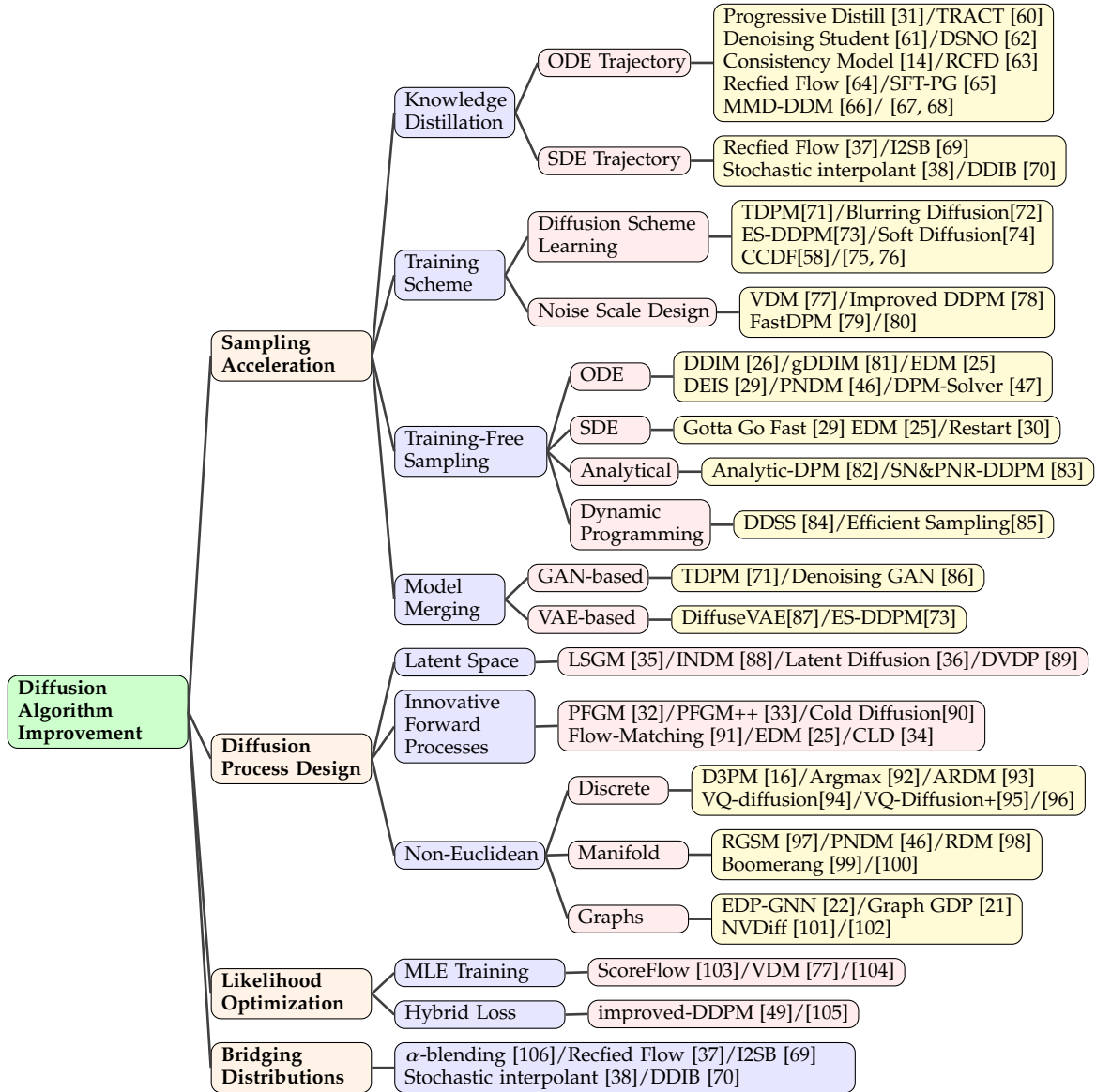
知识提炼是一种将“知识边缘”从更大模型转移到更简单模型的技术，正变得越来越流行 [107, 108]。在扩散模型中，目标是通过对齐和最小化原始样品和生成样品之间的差异，使用更少的步骤或更小的网络来生成样品。蒸馏被视为跨分布的轨迹优化，为经济高效且更快的可控生成提供了最佳映射。

ODE 轨迹 使用 ODE 公式从教师到学生模型的知识蒸馏，通过跨分布场的有效路径将先验分布映射到目标分布。[31] 首先将这一原理应用于改进扩散模型，通过逐步提炼样本轨迹，每两步拉直一次潜在映射。TRACT [60]、去噪学生 [61] 和一致性模型 [14] 扩展了这种效果，通过在 T 时间直接估计来自噪声样本的干净数据，将加速度提高到 64 和 1024。

通过最佳转运可以获得最佳轨迹 [109]。通过流动匹配最大限度地减少分配之间的运输成本，ReFlow [64] 和 [67] 实现了一步生成。DSNO [62] 提出了一种用于直接时间路径建模的神经运算符。一致性模型 [14]、SFT-PG [65] 和 MMD-DDM [66] 分别使用 LPIPS、IPA 和 MMD 搜索理想轨迹。

SDE 轨迹 蒸馏随机轨迹仍在挑战中。提出的工作很少（参见第 3.4 节）。

表 1 扩散算法改进的分类法 (第 3 节)



3.1.2 Training Schedule

改进培训计划涉及修改独立于抽样的传统培训设置，例如扩散方案和噪声方案。最近的研究强调了训练方案中影响学习模式和模型性能的关键因素。在本小节中，我们将训练增强分为两个主要领域：扩散方案学习和噪声标度设计。

扩散方案 学习 扩散模型将数据投射到变分自动编码器（VAE）等潜在空间中，由于其更高的表现力而更加复杂。这些模型中的反向解码方法可以分为两种方法：编码度优化和投影方法。

编码度优化方法，如 CCDF [58] 和 Franzese 等人 [75]，通过将扩散步骤的数量视为

变量。截断是另一种方法，它通过一步步地从扩散较少的数据中采样来平衡生成速度和样本保真度。TDPM [71] 和 ES DPM [73] 使用 GAN 和 CT 截断法 [110]。投影方法，如软扩散 [74] 和模糊扩散模型 [72]，使用线性损坏（如模糊和蒙版）来探索扩散内核的多样性。

噪声标度设计 在传统的扩散过程中，每个过渡步骤都由注入的噪声决定，这相当于在正向和反向轨迹上随机行走。设计噪声标度可以导致合理的生成和快速收敛。与传统的 DDPM 不同，现有方法在整个过程中将噪声标度视为可学习的参数。

像 VDM [77] 这样的前向噪声设计方法将噪声标度参数化为信噪比，将其与训练损失和模型类型联系起来。FastDPM [79] 使用离散时间变量将噪声设计与 ELBO 优化联系起来

或方差标量。对于反向噪声设计，改进的DDPM [78]通过训练混合损失来隐式学习反向噪声尺度，而San Roman *et al.* 使用噪声预测网络在祖先采样之前更新反向噪声尺度。

3.1.3 Training-Free Sampling

免训练方法旨在利用高级采样器来加速预训练扩散模型的采样过程，无需重新训练模型。本小节将这些方法分为几个方面：扩散 ODE 和 SDE 采样器的加速、分析方法和动态规划。

ODE 加速 [11] 表明 DDPM 中的随机采样过程具有边际等效概率 ODE，它定义了数据分发之前的确定性采样轨迹。鉴于 ODE 采样器产生的离散化误差比随机采样器 [11, 30] 少，以前大多数关于采样加速的工作都是以 ODE 为中心的。例如，广泛使用的采样器 DDIM [26] 可以被视为概率流 ODE [11]：

$$d\bar{x}(t) = \epsilon_{\theta}^{(t)} \left(\frac{\bar{x}(t)}{\sqrt{\sigma_t^2 + 1}} \right) d\sigma(t) \quad (15)$$

其中 σ_t 由 $\sqrt{1 - \alpha_t}/\sqrt{\alpha_t}$ 参数化， $\bar{x}$ 参数化为 $x/\sqrt{\alpha_t}$ 。后来的工作 [29, 47] 将 DDIM 解释为对方差保持 (VP) 扩散的 ODE 应用指数积分器的产物 [11]。高级 ODE 求解器已用于 PNDM [46]、EDM [25]、DEIS [29]、gDDIM [81] 和 DPM-Solver [47] 等方法。例如，EDM 采用 Heun 的 2nd 阶 ODE 求解器，而 DEIS/DPM-solver 通过在每个离散时间间隔内对分数函数进行数值近似来改进 DDIM。与原始 DDPM 采样器相比，这些方法显著加快了采样速度（减少了功能评估或 NFE 的数量），同时仍能产生高质量的样本。

基于 SDE 加速 ODE 的采样器速度更快，但达到性能极限，而基于 SDE 的采样器尽管速度较慢，但提供更好的采样质量。一些工作专注于加快随机采样器的速度。Gotta Go Fast [11] 使用自适应步长进行更快的 SDE 采样，而 EDM [25] 将高阶 ODE 与类似 Langevin 动力学的噪声添加和去除相结合，表明他们提出的随机采样器明显优于 ImageNet-64 上的 ODE 采样器。最近的一项工作 [30] 表明，尽管 ODE 采样器涉及较小的具体化误差，但 SDE 中的随机性有助于收缩累积误差。这导致了 Restart Sampling 算法 [30]，它融合了两个世界的最佳方面。采样方法交替通过额外的向前步骤添加显著的噪声，并严格遵循向后 ODE，在标准基准和稳定扩散模型 [36] 上超越了以前的 SDE 和 ODE 样本，两者均在

分析方法：现有的免训练采样方法将反向协方差量视为手工制作的噪声序列，而不动态考虑它们。从 KL 散度优化开始，分析方法使用 Monte Carlo 方法设置反向均值和协方差。

Analytic-DPM [82] 和扩展 Analytic-DPM [83] 共同提出了每种状态校正下的最佳逆解。分析方法对近似误差享有理论保证，但由于其预设，它们仅限于特定分布。

动态规划调整 动态规划 (DP) 通过使用内存化技术 [112] 实现了所有选择的遍历，以在更短的时间内找到最优的解决方案。假设从一个状态到另一个状态的每条路径与其他路径具有相同的 KL 散度，动态规划算法会探索沿轨迹的最佳遍历。当前基于 DP 的方法 [85, 113] 通过优化 ELBO 损失的总和来消耗 $O(T^2)$ 的计算成本。

3.1.4 Merging Diffusion and Other Generative Models

扩散模型可以与生成对抗网络 (GAN) 或变分自动编码器 (VAE) 等其他生成模型协同作用，以简化采样过程。例如，在扩散采样过程的中间阶段，可以通过从噪声样本中获得的 VAE [87] 或 GAN [86] 直接预测原始数据 x_0 。此外，VAE [73] 或 GAN [71] 可以在中间扩散时间步长生成样本，然后由扩散模型对其进行去噪，直到时间 $t = 0$ ，以实现更快的时间遍历。

3.2 扩散工艺设计

扩散模型中的传统正向过程，通常被认为是像素空间中的布朗运动 [10, 25]，对于生成建模来说可能是次优的。因此，研究工作已转向创建新的扩散过程，以简化和增强神经网络的关联后向过程。这条路径已经分叉为开发为扩散模型设计的潜在空间 (第 3.2.1 节) 和用像素空间的改进版本取代直接前向过程 (第 3.2.2 节)。还特别注意了专门为非欧几里得空间定制的扩散过程，如流形、离散空间、泛函空间和图 (第 3.2.3 节)。

3.2.1 Latent Space

研究人员探索在学习的潜在空间中训练扩散模型，以增强神经网络并建立更直接的向后过程。这种方法以 LSGM [35] 和 INDM [114] 为例，它们联合训练了差异融合模型和 VAE 或归一化流模型。两个模型都有一个共同的目标，即加权去噪分数匹配损失 (方程 (13) 中的 L_{DSM})，以优化编码器-解码器和扩散模型对。

$$L := L_{Enc}(z_0|x) + L_{Dec}(x|z_0) + L_{DSM} \left(\left(\{z_t\}_{t=0}^T \right) \right) \quad (16)$$

这里， z_0 表示原始数据 x 的潜在形式，而 z_t 是它的扰动对应物。需要注意的是， z_t 是编码器的一个函数，因此 L_{DSM} 损失也会更新编码器的参数。共同目标是优化 ELBO 或对数似然 [35, 114]。这导致了一个更容易学习和采样的潜在空间。有影响力的工作，如 Stable Diffusion [36] 将

过程分为两个阶段：学习 VAE 的潜在空间和以文本作为条件输入训练扩散模型。另一方面，DVDP [89] 将像素空间分解为正交分量，并在图像扰动期间动态调整每个分量的衰减，类似于动态图像下采样和上采样。

3.2.2 Emerging Forward Processes

潜在空间扩散具有优点，但也增加了框架的复杂性和计算负载。为了解决这个问题，当代研究探索了更健壮和更精细的生成模型的正向过程设计。例如，泊松场生成模型（PFGM）[32] 将数据视为增强空间中的电荷，引导沿电场线的简单分布朝向数据分布。该模型中的正向过程是在电场线的方向上定义的，与扩散模型相比，表现出更稳健的反向采样。PFGM++ [33] 通过更高维的增强变量扩展了 PFGM，这些模型之间的插值揭示了一个最佳点，从而实现了最先进的图像生成。PFGM 和 PFGM++ 也可用于抗体 [115] 和医学图像 [116] 生成。Dockhorn 等 [34] 引入了临界阻尼朗之万扩散（CLD）模型，该模型结合了“车辆”变异

其他研究探讨了替代的腐败过程。例如，冷扩散 [90] 使用任意图像变形（如模糊）进行正向过程，而 [118] 在像素空间中应用散热。此外，人们还努力使用高级高斯微扰核来增强训练和采样 [25, 91]。

3.2.3 Diffusion Models on non-Euclidean space

离散空间 Deep 生成模型在各个领域都取得了显著的进步，例如自然语言处理 [119, 120]、多模态学习 [49, 121] 和人工智能科学 [122, 123]。一项关键成就是处理离散数据，包括句子、残基、原子和向量量化数据。扩散模型通常用于这些应用程序，侧重于文本、分类数据和向量量化数据。D3PM [16] 定义了离散空间中的正向过程，使用过渡内核 Q_t 处理文本或原子类型等数据：

$$q(x_t | x_{t-1}) = \text{Cat}(x_t; p = x_{t-1} Q_t) \quad (17)$$

其中 $\text{Cat}()$ 表示分类分布。此应用已扩展到生成语言文本、分割映射和无损压缩 [92, 93]。

对于文本到图像生成和文本到 3D 生成等多模态问题，矢量量化（VQ）数据将数据转换为代码，在自回归编码器中实现优异的性能 [124]。[94] 首先将扩散技术应用于 VQ 数据，解决了 VQ-VAE 中的单一偏差和累积预测误差。

这一核心思想已被用于进一步的文本到图像、文本到姿势和文本到多模态的工作中 [95, 125–129]。正向过程由概率转移矩阵 Q 和分类表示向量 v 定义：

$$q(x_t | x_{t-1}) = v^\top(x_t) Q_t v(x_{t-1}) \quad (18)$$

流形图像和视频等数据结构通常位于欧几里得空间中。然而，机器人学 [130]、地球科学 [131] 和蛋白质建模 [132] 等领域的某些数据是在黎曼流形 [133] 中定义的。标准 Euclidean 方法可能不适用于此环境。为了解决这个问题，最近的方法，如 RDM [98]、RGSM [97] 和 Boomerang [99] 已将扩散样本纳入黎曼流形，扩展了评分 SDE 框架 [11]。理论著作 [46, 100] 为流形采样提供了进一步的支持。

基于图形的神经网络因其在人体姿势 [127]、分子 [134] 和蛋白质 [135] [136] 中的表现力而越来越受欢迎。当前方法将差异融合理论应用于图形。EDP-GNN [22]、Pan *et al.* [102] 和 GraphDP [21] 等方法通过邻接矩阵处理图数据，以捕获排列不变性。NVDiff [101] 使用反向 SDE 重建节点位置。Function Dutoroir *et al.*, [137] 引入了功能空间中的第一个扩散模型采样，通过联合后验采样捕获无限维分布。

3.3 似然优化

虽然扩散模型 [10] 优化了 ELBO 以克服对数似然的棘手性，但忽略了似然优化，这对于连续时间扩散模型来说是一个挑战 [11]。两种方法，包括 MLE 训练（第 3.3.1 节）和混合损失（第 3.3.2 节）旨在增强可能性训练。

3.3.1 MLE Training

三个同时进行的工作——ScoreFlow [103]、VDM [77] 和 [104] 主要通过使用 Girsanov 定理，在 MLE 训练和扩散模型中的加权去噪分数匹配（DSM）对象之间建立了联系。例如，ScoreFlow [103] 表明，在特定的加权方案下，DSM 目标提供了负对数似然的上限。这一发现使基于分数的 MLE 的神经网络参数无关近似成为可能。

3.3.2 Hybrid Loss

某些方法不仅仅依赖于最大似然训练，而是引入了混合损失设计来提高 DSM 中的模型似然。其中一种方法是 Improved DDPM [78]，它建议使用简单的重新参数化技术和结合变分下界和 DSM 的混合学习目标来学习反向过程的方差。此外，[105] 表明，结合高阶分数匹配损失有助于提高对数似然。

3.4 桥接发行版

扩散模型擅长转换简单的高斯分布，但在桥接任意分布时面临挑战，特别是在图像到图像的转换和细胞分布传输等领域。已经提出了各种方法来解决这个问题。一种方法称为 α 混合 [106]，涉及迭代混合和解混合，以创建一个确定性的桥梁。当一端分布为高斯分布时，扩散模型被视为特殊情况。另一种方法是整流 [37]，它结合了额外的步骤来拉直桥梁。其他方法，例如 [38] 中提出的方法，建议在两个分布之间构建一个具有一般插值函数的 ODE。此外，其他人探索了利用 Schrödinger 桥 [69] 或高斯分布作为连接点来连接两个扩散常微分方程 [70]。

4 应用

得益于生成真实样品的强大能力，扩散模型已广泛应用于各个领域。在实际应用中，释放扩散模型功能的关键在于将扩散过程、去噪过程和条件采样拟合到各种数据的自然值。受这个想法的启发，扩散的应用被总结为图像生成、3D 生成、视频生成、医学分析、文本生成、时间序列生成、音频生成、分子设计和图形生成。

4.1 图像生成

扩散模型在图像生成方面取得了显著的性能，无论是在传统的类条件或无条件生成 [10,27,138]，还是在更复杂的文本或图像条件 [36,143]，或它们的组合 [142]。今后，我们的讨论将集中在模拟真实场景的应用程序设置上，根据条件输入对应用程序进行分类。

4.1.1 Text condition

扩散模型在文本到图像生成方面表现出卓越的性能，不仅能够创建逼真图像，还能够创建与用户提供的文本输入紧密相关的样本。突出的例子包括 Imagen [138]、稳定扩散 [36] 和 DALL-E 2 [121]。这些方法建立在现有的扩散架构之上，增加了一个交叉注意力层，将文本嵌入序列注入扩散模型。实验结果表明，这种条件机制有效地将文本信息混合到生成的图像中。

此外，交叉注意力条件机制通过利用和操作交叉注意力层中的键、值或注意力矩阵，实现了许多免训练的图像编辑。例如，[139] 通过在交叉注意力层的输出中交换或添加新的特征图来改变源图像中的感知；[140] 通过学习新的文本嵌入作为交叉注意力层的输入，实现新概念的定制。[141] 强制交叉注意力文本提示中的所有子标记并放大它们的激活，鼓励模型忠实地生成文本提示中描述的所有主题。

4.1.2 Image condition

除了文本条件外，扩散模型还支持图像条件，例如要编辑的图像、深度图或人体骨骼，作为条件输入。基本概念保持不变，涉及将编码的图像特征合并到扩散主干中。[142] 的工作将源图像中的编码特征引入第一个卷积层，以实现图像调节，从而允许使用文本提示进行图像到图像的编辑。同样，[143] 利用深度图、Canny 边缘或人类骨骼来控制生成图像的空间布局。

4.2 3D 生成

从广义上讲，通过扩散模型生成 3D 有两种主要方法。第一种方法侧重于直接使用 3D 数据训练这些模型。然而，由于 3D 数据的可用性有限，第二种方法强调通过 2D 扩散先验生成 3D 内容的大小。

4.2.1 3D Data Condition

鉴于 3D 表示的多样性，例如 NeRF、点云、体素、高斯飞溅等，扩散模型已有效地应用于这些不同的 3D 表示。例如，[147, 150, 151] 等作品直接为 3D 对象生成点云。为了实现高效的采样，在 PDR 中采用混合点体素表示进行形状处理 [144]，引入了点云完成的新范式。在这项研究的基础上，Point-E [146] 进一步将图像合成作为点云扩散模型的附加条件输入。

相比之下，Shape-E [145] 利用扩散模型来表示 3D 物体的 NeRF。Zero-1-to-3 [148] 采用不同的方法，通过训练视点条件的差异融合模型来实现新的视图合成。然后，它根据从不同相机视点生成的样本优化 NeRF。基于这项工作，[149] 通过结合姿态估计阶段进一步扩展了 0-1-to-3。

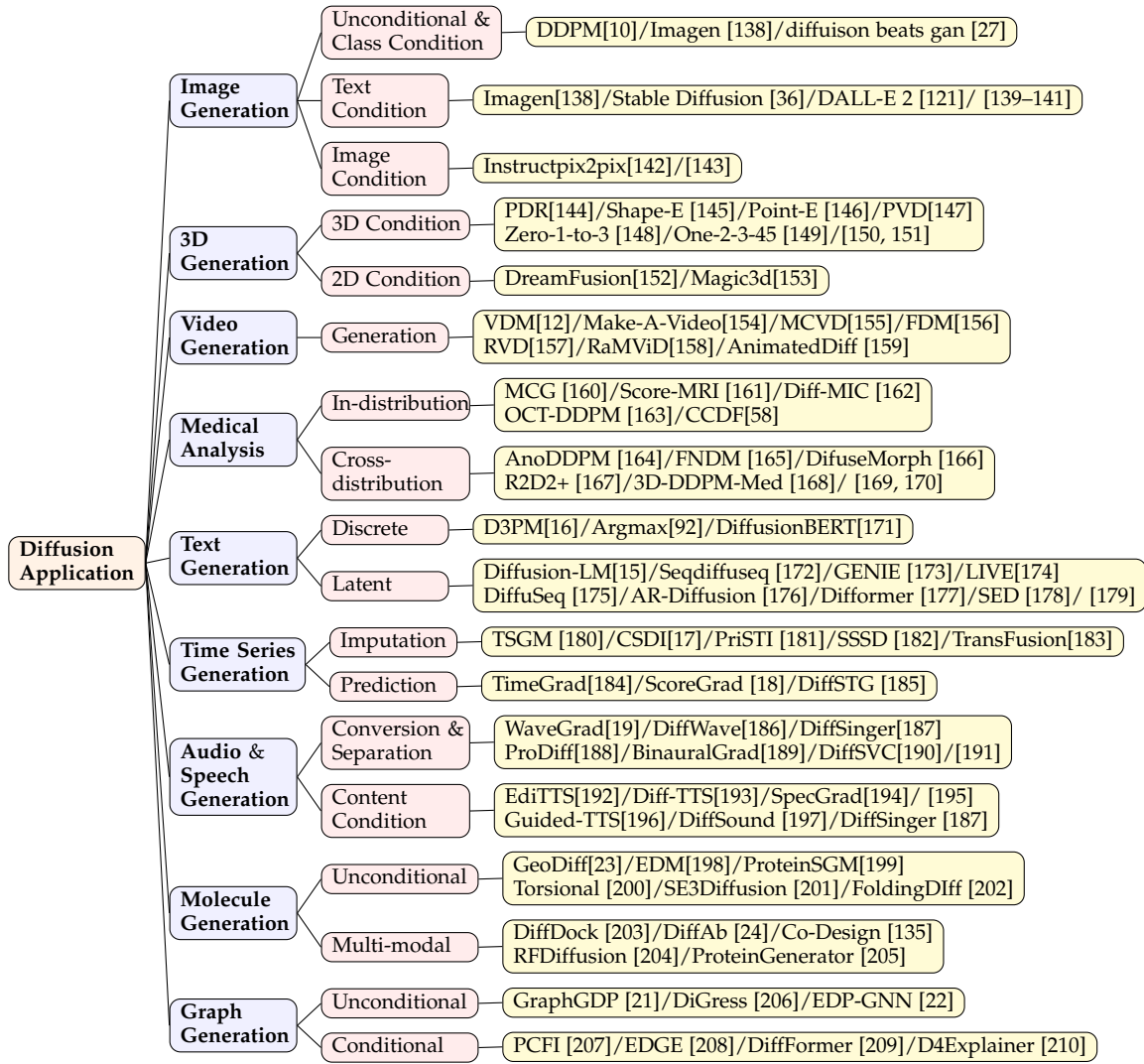
4.2.2 2D Diffusion prior

另一个有趣的工作方向是 *distill 3D from a 2D diffusion model*。Dreamfusion [152] 巧妙地使用分数蒸馏采样 (SDS) 目标，从预先训练的文本到图像模型中提取 NeRF。他们通过梯度下降优化随机初始化的 NeRF，以便从不同角度渲染的图像实现低损耗。[153] 将 DreamFusion 扩展到从粗到精的两阶段运算框架，以加速生成过程。

4.3 视频生成

视频扩散模型通过额外的时间轴增强了用于图像生成的 2D 扩散模型。一般的思路是添加一个时间层，以显式模拟现有 2D 扩散结构中的跨帧依赖性。代表作品包括视频扩散模型 [12]、Make-A-Video [154]、AnimatedDiff [159]、RVD [157]、FDM [156]、MCVD [155]。RaMViD [158] 将图像扩散模型扩展到具有 3D 卷积神经网络的视频，并设计了一种用于视频预测、填充和上采样的调节技术。

TABLE 2
Classification of Diffusion-based model Applications(Section 4)



4.4 医学分析

扩散模型为医学分析中面临的挑战提供了一种解决方案，其中获取大规模、高质量的注释数据集具有挑战性。这些模型在与分布内分析和交叉分布生成相关的任务中表现出卓越的性能。

4.4.1 In-distribution Analysis

扩散模型在各种医学成像任务中都很有效，利用它们准确捕获具有强大先验信息的医学图像的能力。它们已成功用于超分辨率 [58, 160]、分类 [162] 和噪声鲁棒性 [161, 163]。例如，Score-MRI [161] 使用像素引导 SDE 采样加速 MRI 重建，而 Diff-MIC [162] 通过双粒度引导和最大均值差异实现跨多种模态的精确分类。此外，MCG [160] 提出了在采样过程中进行歧管校正以实现 CT 超分辨率，从而减少误差并提高加速度。

4.4.2 Cross-distribution Generation

多模态制导显著提高了医学分析的生成能力。通过整合特定类的指导 [164, 165] 和像素级指导 [165, 166, 169]，无条件去噪网络可以在不同类型的稀缺图像之间进行图像转换，包括高质量格式图像、健康图像和无偏图像。值得注意的例子包括 FNDM [165]，它能够通过具有混合条件指导的非马尔可夫框架准确检测大脑异常，以及 DiffuseMorph [166]，它使用以移动和固定图像对为条件的连续扩散采样进行 MR 图像配准。此外，有一些很有前途的方法可以用从少量高质量样本生成的逼真医学图像来丰富训练数据集 [167, 168, 170]。例如，在 31,740 个样本上训练的潜在扩散模型已被用于合成具有 100,000 个实例的高质量 and 语义丰富的数据集。

4.5 文本生成

文本生成通过生成自然和连贯的语言，在弥合人类和高级人工智能之间的差距方面发挥着至关重要的作用。自回归语言模型按顺序生成文本，确保高语义连贯性，但生成速度较慢 [211]。另一方面，扩散模型支持并行文本生成，提供更快的速度，但相对较弱的语义连贯性 [212, 213]。两种主要方法，即离散生成和潜在生成，通常用于解决生成离散令牌的挑战。

4.5.1 Discrete Generation

离散生成方法涉及将离散词作为输入的模型，并利用先进的技术、参数化和预训练模型。开创了扩散模型和离散生成之间的连接，包括 D3PM [16] 和 Argmax [92] 在内的典型工作将单词视为分类向量。他们使用离散的转换矩阵建立前向和后向过程，将要生成的数据视为平稳分布。DiffusionBERT [171] 将扩散模型与预先训练的语言模型相结合，展示了改进的文本生成性能。此外，它还引入了一种新的噪声调度，并探索了将时间步长纳入 BERT 以进行反向扩散过程。

4.5.2 Latent Generation

第二种方法侧重于在标记的潜在空间中生成文本，捕捉扩散过程的连续性。它结合了增强的损失功能 [172, 176, 177]、多样化的代表类型 [175, 178] 和先进的模型架构 [15, 173]。例如，LM-Diffusion [15] 引入了基于 transformer 的图形模型进行可控生成，在各种文本生成任务中表现出卓越的性能。GENIE [173] 提出了一种基于扩散框架的大规模语言模型，该模型结合了一种新的连续参数图降噪（CPD）损失，以改善去噪和段落级连贯性。它展示了基于扩散的解码器在文本生成方面的潜力，并为未来的研究提供了坚实的基础。除了高级条件采样、标记级捕获和后细化之外，NLP 中的扩散模型有望增强嵌入空间的建模 [178, 213]，建立与 larg 的联系

4.6 时间序列生成

准确的时间序列建模对于趋势预测、决策和实时分析至关重要。扩散模型通过时间序列数据模块增强了这一过程，实现了卓越的分析和多样化的生成 [214]。根据不同类型的掩蔽策略，先验条件可以分为掩蔽任务和预测任务。在修复任务中，观察到的状态被用作先验条件 [17, 180–182]，与基于上下文的模块相结合。CSDI 提出了一种基于双向 CNN 模块的自监督训练框架，

在医疗保健和环境数据的连续生成方面取得实质性改进 [17]。对于预测任务，先验状态被转化为用户定义的特征和潜在嵌入，作为自我条件 [18, 184, 185]。结合时空模块，如图 UNet 和 RNN，DiffSTG 和 TimeGrad 成功地实现了时间序列的时空概率学习 [184, 185]。时间序列生成的成功取决于时间相关序列的准确建模以及在采样过程中加入稳健的自条件指导。这些方面指向该领域有希望的未来进展 [183, 185, 214]。

4.7 音频生成

合成高质量的模拟语音在音乐创作、虚拟现实、游戏开发和语音助手方面有多种应用，提供个性化和身临其境的音频体验，并改善人机交互。扩散模型非常适合处理音频数据的独特特征，它利用强先验并有效管理高维、时间依赖性信息。语音生成依赖于混合条件，将文本和控制标签相结合，以实现特定的语义或声音特征。WaveGrad [19]、DiffSinger [187] 和 DiffSVC [190] 等技术使用梅尔三维频谱图作为条件指导，而 BinualGrad [189] 则根据单声道音频输入分离音频。这些方法构成了一般波形生成的基础，而响度、旋律和语音后序等附加功能可实现可控的风格生成 [187, 190, 191, 215]。基于文本和基于音乐的生成，包括文本到语音转换

4.8 分子设计

分子作为生命的基本组成部分，在许多生物过程中起着至关重要的作用。长期以来，功能分子的设计一直是一个具有挑战性和持久的问题 [216]。生成模型通过为传统、费力的计数和实验验证方法提供更有效的替代方案，彻底改变了分子设计。通过表征特定的模态分布和功能域，生成模型可以产生新颖有效的药物分子结构，从而扩展了药物设计的可能性 [217]。在药物发现领域，扩散模型有效地探索了广阔的化合物空间，加快了对潜在候选药物的搜索。这提高了药物发现过程的整体效率，并揭示了错综复杂的化合物研究，有助于更好地了解药物机制。在分子设计中观察到的模式大致可分为无条件生成和 c

4.8.1 Unconditional Generation

无条件分子生成侧重于使用扩散模型生成分子结构，该模型具有速度和高质量的建模能力。一种方法是生成分子在三维空间中的位置，捕获分子在空间中的构象 [23, 135, 198]。然而，由于分子三维结构的分布不均匀和不规则，这种方法可能会导致较低多样性和较大的误差。或者，生成捕获多个特征的模型以及高维空间中结构特征的分布可以导致更多样化的分布和可解释性 [199–202]。[218] 进一步引入了样本之间的排斥力以促进多样性。

4.8.2 Cross-modal Generation

在分子设计中，跨模态生成侧重于将功能作为条件。基于扩散的方法擅长结合条件并利用基于不同模态的去噪模型来增强建模能力。基于序列的跨模式生成方法利用蛋白质序列和多序列比对 (MSA) 序列来训练去噪模型，结合特定的蛋白质结构信息和功能标记来指导生成 [205, 219]。基于结构的跨模态方法利用结构预测模型的先验知识来协助精确引导生成，结合蛋白质序列和功能信息 [204]。分子对接和抗体设计方法利用焦油分子的结构先验来指导对接过程并确定有利的结合构型 [24, 203]。这些方法利用目标结构的先验知识来增强生成和 o

4.9 图形生成

使用扩散模型生成图形的动机源于研究和模拟各种真实世界网络和传播过程的目标。通过这样做，它为实际问题提供了更好的理解和问题解决能力。这种方法使研究人员能够深入研究复杂系统内的相互作用和信息传播机制，揭示隐藏的模式和相关性，并能够预测潜在结果。该方法的应用包括社交网络分析、生物神经系统分析以及图形数据集的生成和评估。在 3.2.3 节中，我们之前提到了传统的图生成方法，该方法涉及通过离散扩散生成邻接矩阵或节点特征 [21, 22, 206]。然而，这些无条件生成的图形具有有限的可扩展性，缺乏实际适用性。因此，占主导地位的 appr

伪自信，作为世代过程中的指导因素。另一方面，EDGE [208] 和 DiffFormer [209] 分别利用节点度和能量约束作为条件，以实现离散和连续生成邻接矩阵和潜在嵌入，从而拓宽了生成可能性的范围。此外，D4Explainer [210] 将图数据的分布作为一个条件，并结合分布损失和反事实损失来探索反事实实例。

5 结论与讨论

5.1 结论

扩散模型在深度学习领域变得越来越重要。为了利用扩散模型的力量，本文通过对各种态度的详细观察，包括基础理论、改进的算法和应用，对扩散模型的几个方面进行了全面和最新的回顾。我们希望这项调查成为读者的综合指南，了解扩散模型增强的进展，并为其实际应用提供有价值的见解。

5.2 与现有调查的比较

在扩散模型领域已经存在几种调查，包括一般调查 [157]、包括视觉 [220]、语言处理 [212, 213]、音频 [221]、时间序列 [214]、医学分析 [222] 和生物信息学 [223, 224]，以及不同数据结构的调查 [225, 226]。与现有调查相比，我们进行了全面的审查，并提供了广泛的见解，包括算法增强和广泛的应用。此外，我们保持该领域的最新更新，以跟踪最新的改进，并每月维护我们的 GitHub 存储库以进行长期分析。

5.3 限制和未来方向

5.3.1 Challenges Under Data Limitations

除了推理速度慢外，扩散模型经常遇到从低质量数据中辨别模式和规律的困难，导致它们无法推广到新的场景或数据集。此外，处理大规模数据集会带来计算挑战，例如训练时间延长、内存使用过多或无法收敛到所需状态，从而限制模型的规模和复杂性。此外，有偏差或不均衡的数据样本会限制模型生成可适应不同领域或人口统计数据的输出的能力。

5.3.2 Controllable Distribution-based Generation

提高模型理解和生成特定分布中样本的能力对于使用有限的数据实现更好的泛化至关重要。通过专注于识别数据中的模式和相关性，该模型可以生成与训练数据紧密匹配并满足特定要求的样本。这需要有效的数据采样、利用技术以及优化模型参数和结构。最终，这种增强的基础允许更可控和更精确的生成，从而提高泛化性能。

5.3.3 Advanced Multi-modal Generation Leveraging LLMs

扩散模型的未来方向需要通过集成大型语言模型（LLM）来推进多模态生成。这种集成使模型能够生成包含文本、图像和其他模态组合的输出。通过整合 LLM，该模型增强了对不同模态之间相互作用的理解，从而产生更加多样化和现实的输出。此外，LLM 通过有效地利用文本和其他模态之间的联系，显著提高了基于提示的生成效率。此外，LLM 还可以作为提高扩散模型生成能力的催化剂，扩大它可以生成模态的领域范围。

5.3.4 Integration with Machine Learning Fields

将扩散模型与传统的机器学习理论相结合，为提高各种任务的性能提供了新的机会。半监督学习在解决扩散模型的固有挑战（如泛化）方面特别有价值，即使在数据有限的情况下也能实现有效的条件生成。通过利用未标记的数据，它增强了扩散模型在特定条件下生成样本时泛化和实现理想性能的能力。

此外，强化学习在模型的采样过程中采用微调算法提供有针对性的指导，从而发挥着至关重要的作用。这种指导确保了集中的探索并促进了受控的生成。此外，结合额外的反馈丰富了强化学习，从而提高了模型的可控制条件生成能力。

确认

本文中描述的这项工作部分得到了中国香港特别行政区研究资助局的资助（项目编号：T45-401/22-N）和香港创新及科技基金的资助（项目编号：ITS/241/21）。这项工作得到了国家重点研发计划（No. 2022YFE0200700）、国家自然科学基金（No. 62006219 and 62376254）、广东省自然科学基金（No. 2022A1515011579）的部分支持。这项工作也得到了科技创新 2030 - 重大项目（编号 2021ZD0150100）和中国国家自然科学基金（编号 U21A20427）的部分支持。

引用

[1] D. P. Kingma, M. Welling *et al.*, “变分自动编码器简介”, *Foundations and Trends® in Machine Learning*, 2019 年。（文件）[2] A. Oussidi 和 A. Elhassouny, “深度生成模型：调查”, *ISCV. IEEE*, 2018 年。（文件）[3] Y. LeCun, S. Chopra, R. Hadsell, M. Ranzato, and F. Huang, “基于能量的学习教程”, *Predicting structured data*, 2006.（文件）

[4] J. Ngiam, Z. Chen, P. W. Koh, and A. Y. Ng, “Learning deep energy models”, in *ICML*, 2011.（文件）[5] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville 和 Y. Bengio, “生成对抗网络”, *Communications of the ACM*, 2020 年。（文件）[6] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta 和 A. A. Bharath, “生成对抗网络：概述”, *IEEE Signal Process*, 2018 年。（文件）[7] D. Rezende 和 S. Mohamed, “具有归一化流的变分推理”, *ICML*, 2015 年。（文件）[8] I. Kobyzev, S. J. Prince 和 M. A. Brubaker, “规范化流：当前方法的介绍和回顾”, *IEEE TPAMI*, 2020 年。（文件）[9] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan 和 S. Ganguli, “利用非平衡热力学的深度无监督学习”, *ICML*, 2015 年。（文档）, 2.1.3, 7 [10] J. Ho, A. Jain, and P. Abbeel, “Denoising diffusion probabilistic models”, *NeurIPS*, 2020.2.1.2, 2.1.3, 2.2.1, 3.2, 3.3, 4.1, 2, 1,

- [21] H. Huang, L. Sun, B. Du, Y. Fu, and W. Lv, "Graphgdp: Generative diffusion processes for permutation invariant graph generation," *arXiv:2212.01842*, 2022. (文档), 1, 3.2.3, 2, 4.9 [22] C. Niu, Y. Song, J. Song, S. Zhao, A. Grover, and S. Ermon, "Permutation invariant graph generation via score-based generative modeling," *AISTATS.PMLR*, 2020 年, 第 4474-4484 页. (文件), 1, 3.2.3, 2, 4.9, 7 [23] M. Xu, L. Yu, Y. Song, C. Shi, S. Ermon, and J. Tang, "Geodiff: A geometric diffusion model for molecular conformation generation," in *ICLR*, 2021. (文件), 2, 4.8.1, 8 [24] S. Luo, Y. Su, X. Peng, S. Wang, J. Peng, and J. 马, "使用基于扩散的蛋白质结构生成模型进行抗原特异性抗体设计和优化," *NeurIPS*, 2022 年. (文件), 2, 4.8.2, 8 [25] T. Karras, M. Aittala, T. Aila, and S. Laine, "阐明基于扩散的生成模型的设计空间," *arXiv:2206.00364*, 2022 年. (文件), 1,3.1.3,3.2,3.2.2,6,7 [26] J. Song, C. Meng, and S. Ermon, "Denoising diffusion," 1, 3.4 [38] M. S. Albergo 和 E. Vanden-Eijnden, "使用随机插值器构建规范化流", *ArXiv*, 卷 abs/2209.15571, 2022 年. (文档), 1, 3.4 [39] L. Arnold, "随机微分方程", *New York*, 1974 年. 2.2.2 [40] B. 奥克森达尔, *Stochastic differential equations: an introduction with applications*. Springer Science & Business Media, 2013 年. 2.2.2 [41] B. D. Anderson, "反时间扩散方程模型", *Stochastic Processes and their Applications*, 第 12 卷, 第 3 期, 第 313-326 页, 1982 年. 2.2.2 [42] P. Vincent, "分数匹配和去噪自动编码器之间的联系", *Neural computation*, 2011 年. 2.2.2 [43] Y. Xu, S. Tong 和 T. Jaakkola, "扩散模型中用于降低方差分数估计的稳定目标区域", *ArXiv*, 卷 abs/2302.00670, 2023 年. 2.2.2, 6 [44] D. Maoutsa, S. Reich, and M. Opper, "通过梯度-对数-密度估计实现福克-普朗克方程的交互粒子解", *Entropy*, 第 22 卷, 第 8 期, 第 802 页, 2020 年. 2.2.2 [45] R. T. Chen, Y. Rubanova, J. Bettencourt, and D. K. Duvenaud, "神经常微分方程", *NeurIPS*, 第 3 卷

outs 和 C. G. Snoek, “自导向扩散模型”, *arXiv:2210.06462*, 2022 年。2.3, 6 [57] C.-H.赵 W.-F.Sun, B.-W.Cheng 和 C.-Y.Lee, “基于准保守分数的生成模型”, *arXiv:2209.12753*, 2022 年。2.3 [58] H. Chung, B. Sim, and J. C. Ye, “Come-closer-diffuse- faster: Accelerating conditional diffusion models for inverse problems through stochastic contraction”, *CVPR*, 2022 年。2.3, 1, 3.1.2, 2, 4.4.1, 7 [59] J. Choi, S. Kim, Y. Jeong, Y. Gwon, and S. Yoon, “I lvr: Denoising diffusion probabilistic models的条件反射方法”, *CVPR*, 2021 年, 第 14 367-14 376 页。2.3 [60] D. Berthelot, A. Autef, J. Lin, D. A. Yap, S. Zhai, S. 胡, D. Zheng, W. Talbot, and E. Gu, “Tract: De-noising diffusion models with transitive closure time- distillation”, *arXiv:2303.04248*, 2023.3.1.1, 1 [61] E. Luhman 和 T. Luhman, “迭代生成模型中的知识提炼以提高采样速度”, *arXiv*, 2021 年。3.1.1, 1, 6, 7 [62] H. Zheng, W. Nie, A. Vahdat, K. Azizzadenesheli, and A. Anandkumar, “快速采样”

生成模型。1, 3.1.2, 6, 7 [76] V. Khrulkov 和 I. Osel edets, “通过最佳传输了解 ddpm 潜在代码”, *arXiv:2202.07477*, 2022 年。1 [77] D. Kingma、T. Salimans、B. Poole 和 J. Ho, “变异扩散模型”, *NeurIPS*, 第 34 卷, 第 21 696-21 707 页, 2021 年。1, 3.1.2, 3.3.1, 5, 7 [78] A. Q. Nichol 和 P. Dhariwal, “改进的去噪扩散概率模型”, *ICML*, 2021 年。1, 3.1.2, 3.3.2, B.3, 5, 6, 7 [79] Z. Kong 和 W. Ping, “关于扩散概率模型的快速采样”, *arXiv:2106.00132*, 2021 年。1, 3.1.2, 6, 7 [80] R. San-Roman, E. Nachmani, and L. Wolf, “生成扩散模型的噪声估计”, *arXiv:2104.02600*, 2021 年。1, 7 [81] Q. Zhang, M. Tao, and Y. Chen, “gddim: 广义降噪扩散隐含模型”, *arXiv:2206.05564*, 2022 年。1, 3.1.3, 6, 7 [82] F. Bao, C. Li, J. Zhu, and B. Zhang, “Analytic-dpm: An analytic estimate of the optimal reverse variance in diffusion probabilistic models”, *arXiv:2201.06503*, 2022.1, 3.1.3, 3, 4, 5, 6, 7 [83] F. Bao, C. Li, J. Sun, J. Zhu, 和 B. Z

- sion 模型 ”, *arXiv:2110.02037*, 2021 年。1, 3.2.3, 7 [94] 顾淑娴、陈大军、包建军、温峰、张斌斌、袁立军和郭斌, “用于文本到图像合成的矢量量化扩散模型”, *CVPR*, 2022 年, 第 10 696-10 706 页。1, 3.2.3, 7, 8 [95] Z. Tang, S. Gu, J. Bao, D. Chen, and F. 温, “改进的矢量量化扩散模型”, *arXiv:2205.16007*, 2022.1, 3.2.3, 7, 8 [96] A. Campbell, J. Benton, V. De Bortoli, T. Rainforth, G. Deligiannidis, and A. Doucet, “离散降噪模型的连续时间框架”, *arXiv:2205.14987*, 2022.1, 7 [97] V. De Bortoli, E. Mathieu, M. Hutchinson, J. Thornton, Y. W. Teh, and A. Doucet, “基于黎曼分数的生成建模”, *arXiv:2202.02763*, 2022 年。1, 3.2.3, 7 [98] C.-W. Huang, M. Aghajohari, A. J. Bose, P. Panangaden, and A. Courville, “黎曼扩散模型”, *arXiv:2208.07949*, 2022.1, 3.2.3, 7 [99] L. Luzi, A. Siahkoohi, P. M. Mayer, J. Casco-Rodriguez, and R. Baraniuk, “回旋镖：使用扩散模型对图像流形进行局部采样”, *arXiv:2210.12100*, 2022 年。1, 3.2.3 [100] X. C [113] D. Watson, W. Chan, J. Ho 和 M. Norouzi, “通过样本质量进行区分来学习扩散模型的快速采样器”, *ICLR*, 2021 年。3.1.3 [114] D. Kim, B. Na, S. J. Kwon, D. Lee, W. Kang, 和 I.-c. Moon, “参数化扩散模型的最大似然训练”, *arxiv*, 2021 年。3.2.1, 3.2.1, 7 [115] C. Huang, Z. Liu, S. Bai, L. Zhang, C. Xu, Z. WANG, Y. Xiang, and Y. Xiong, “Pf-abgen: A reliable and efficient antibody generator via poisson flow”, in *ICLR MLDD workshop*, 2023.3.2.2 [116] R. Ge, Y. He, C. Xia, Y. Chen, D. Zhang, and G. Wang, “Jccs-pfgm: 一种基于脑圈监督的新型泊松流生成模型, 用于关节条件下的多相 cect 进行性低剂量重建”, 2023 年。3.2.2 [117] Z. Liu, D. Luo, Y. Xu, T. Jaakkola, and M. Tegmark, “Genphys: From physical processes to generative models”, *ArXiv*, vol. abs/2304.02637, 2023.3.2.2 [118] S. Risannen, M. Heinonen 和 A. Solin, “具有逆散热的生成建模”, *ArXiv*, 第 abs/2 卷

- 一个扩散模型”，*arXiv:2211.08332*，2022 年。3.2.3 [130] H. A. Pierson 和 M. S. Gashler，“机器人技术中的深度学习：近期研究综述”，*Advanced Robotics*，2017 年。3.2.3 [131] R. P. De Lima、K. Marfurt、D. Duarte 和 A. Bonar，“地球科学图像深度学习分析的进展和挑战”，*81st EAGE Conference and Exhibition 2019*。欧洲地球科学家与工程师协会，2019 年。3.2.3 [132] J. Wang、H. Cao、J. Z. Zhang 和 Y. Qi，“使用深度学习神经网络进行计算蛋白质设计”，*Scientific reports*，2018 年。3.2.3 [133] W. Cao，Z. Yan，Z. 他和 Z. 他，“几何深度学习的全面调查”，*IEEE Access*，2020 年。3.2.3 [134] H. Lin、Y. Huang、M. Liu、X. Li、S. Ji 和 S. Z. Li，“Diffbp：用于焦油蛋白结合的 3d 分子的生成扩散”，*arXiv preprint arXiv:2211.11214*，2022 年。3.2.3 [135] N. Anand 和 T. Achim，“等变去噪差异融合概率模型的蛋白质结构和序列生成”，*arXiv:2205.15019*，2022 年。3.2.3，2，4.8.1，8 [136] L. Wu，H. Lin，Z. Gao，C. Tan，and S. Z. Li，“自我监督
- [146] A. Nichol、H. Jun、P. Dhariwal、P. Mishkin 和 M. Chen，“Point-e：从复杂提示生成 3D 点云的系统”，*arXiv preprint arXiv:2212.08751*，2022 年。4.2.1，2 [147] L. 周、Y. Du 和 J. Wu，“通过点体素扩散生成和完成 3D 形状”，*ICCV*，2021 年，第 5826-5835 页。4.2.1，2，8 [148] R. Liu，R. Wu，B. Van Hoorick，P. Tokmakov，S. Zakharov，and C. Vondrick，“Zero-1-to-3：Zero-shot one image to 3d object”，in *ICCV (ICCV)*，2023 年 10 月，pp. 9298–9309.4.2.1，2 [149] M. Liu，C. Xu，H. Jin，L. Chen，Z. Xu，H. Su *et al.*，“One-2-3-45：Any single image to 3d mesh in 45 seconds without per-shape optimization”，*arXiv preprint arXiv:2306.16928*，2023.4.2.1，2 [150] S. Luo 和 W. 胡，“基于分数的点云降噪”，*ICCV*，2021 年，第 4583-4592 页。4.2.1，2，8 [151] ——，“用于 3d 点云生成的扩散概率模型”，*CVPR*，2021 年，第 2837-2845 页。4.2.1，2，7，8 [152] B. Poole，A. Jain，J. T. Barron，and B. Mildenhall，“Dreamfusion：Text-to-3d using 2d diffusion”，*arXiv:2209.14988*

- 在 *Medical Imaging 2022: Image Processing*, 第 12032 卷中。SPIE, 2022 年, 第 25-34 页。2, 4.4.1 [164] J. Wyatt, A. Leach, SM Schmon 和 CG Willcocks, “Anoddpm: 使用单纯形噪声的去噪差异融合概率模型的异常检测”, *CVPR*, 2022 年, 第 650-656 页。2, 4.4.2 [165] J. Li, H. Cao, J. Wang, F. Liu, Q. Dou, G. Chen, 和 P.-A. Heng, “用于大脑 mr 图像中弱监督异常检测的快速非马尔可夫扩散模型”, *MICCAI*. 施普林格, 2023 年, 第 579-589 页。2, 4.4.2 [166] B. Kim, I. Han, and J. C. Ye, “Diffusemorph: unsupervised deformable image registration using diffusion model”, *ECCV*. 施普林格, 2022 年, 第 347-364 页。2, 4.4.2 [167] H. Chung, E. S. Lee, and J. C. Ye, “使用正则化反向扩散的先生图像去噪和超分辨率”, *IEEE Transactions on Medical Imaging*, 第 42 卷, 第 4 期, 第 922-934 页, 2022 年。2, 4.4.2, 8 [168] Z. Dorjsembe, S. Odonchimed, and F. Xiao, “使用去噪扩散概率模型的三维医学图像合成”, *Medical Imaging with Deep Learning*, 2022 年。2, 4.4.2 [169] 龚淑娴, 陈春, Y [178] R. Strudel, C. Tallec, F. Altché, Y. Du, Y. Ganin, A. Mensch, W. Grathwohl, N. Savinov, S. Dieleman, L. Sifre *et al.*, “用于文本生成的自条件嵌入扩散”, *arXiv preprint arXiv:2211.04236*, 2022 年。2, 4.5.2 [179] J. Ye, Z. Zheng, Y. Bao, L. Qian, and Q. Gu, “Diffusion language models can perform many tasks with scaling and instruction-finetuning”, *arXiv preprint arXiv:2308.12219*, 2023.2, 4.5.2 [180] H. Lim, M. Kim, S. Park, and N. Park, “使用 SGM 生成规则时间序列”, *arXiv preprint arXiv:2301.08518*, 2023 年。2, 4.6 [181] M. Liu, H. Huang, H. Feng, L. Sun, B. Du, and Y. Fu, “Pristi: A conditional diffusion framework for spatiotemporal imputation”, *arXiv preprint arXiv:2302.09746*, 2023.2 [182] J. M. Lopez Alcaraz 和 N. Strodthoff, “使用结构化 atate space 模型进行基于扩散的时间序列插补和预测”, *Transactions on machine learning research*, 第 1-36 页, 2023 年。2, 4.6 [183] M. Fahim Sikder, R. Ramachandranpillai, and F. Heintz, “输血: 使用带有转化器的扩散模型生成高保真的时间序列”, *arXiv e-prints*,

- [194] Y. Koizumi, H. Zen, K. Yatabe, N. Chen, and M. Bacchiani, “Specgrad: 基于扩散概率模型的神经声码器, 具有自适应噪声频谱整形”, *arXiv:2203.16749*, 2022 年。2, 4.7, 8 [195] S. Kim, H. Kim, and S. Yoon, “Guided-tts 2: A diffusion model for high-quality adaptive text-to-speech with untranscribed data”, *arXiv:2205.15370*, 2022.2, 4.7, 8 [196] H. Kim, S. Kim, and S. Yoon, “Guided-tts: A diffusion model for text-to-speech via classifier guidance”, *ICML. PMLR*, 2022 年, 第 11、119-11、133 页。2, 4.7, 8 [197] D. Yang, J. Yu, H. Wang, W. Wang, C. Weng, Y. Zou, and D. Yu, “Diffsound: 用于文本到声音生成的离散扩散模型”, *arXiv:2207.09983*, 2022 年。2, 4.7, 8 [198] E. Hogeboom, V. G. Satorras, C. Vignac, and M. Welling, “三维分子生成的等变扩散”, *ICML.PMLR*, 2022 年, 第 886 7-8887 页。2, 4.8.1, 8 [199] J. S. Lee 和 P. M. Kim, “ProteinSGM: 基于分数的从头蛋白质设计生成建模”, *bioRxiv*, 2022 年。2, 4.8.1, 8 [200] B. Jing, G. Corso, R. Barzilay [210] J. Chen, S. Wu, A. Gupta, and R. Ying, “D4explainer: In-distribution gnn explanations via discrete denoising diffusion”, *arXiv preprint arXiv:2310.19321*, 2023.2, 4.9 [211] D. W. Otter, J. R. Medina 和 J. K. Kalita, “深度学习在自然语言处理中的用途调查”, *IEEE transactions on neural networks and learning systems*, 第 32 卷, 第 2 期, 第 604-624 页, 2020 年。4.5 [212] H. Zou, Z. M. Kim 和 D. Kang, “NLP 中的扩散模型: 一项调查”, *arXiv preprint arXiv:2305.14671*, 2023 年。4.5, 4.5.2, 5.2 [213] 李彦, 周国, 赵伟X, 和 J.-R.温, “非自回归文本生成的扩散模型: 调查”, *arXiv preprint arXiv:2303.06574*, 2023 年。4.5, 4.5.2, 5.2 [214] L. Lin, Z. Li, R. Li, X. Li, and J. Gao, “时间序列应用的扩散模型: 一项调查”, *arXiv preprint arXiv:2305.00624*, 2023 年。4.6, 5.2 [215] V. Popov, I. Vovk, V. Gogoryan, T. Sadekova, M. S. Kudinov, and J. Wei, “基于扩散的语音转换与快速最大似然采样方案”, *ICLR*, 2021 年。4.7, 8 [216] S. Min, B. Lee, and S. Yoon, “生物信息学中的深度学习”, *Briefings in bioinformatics*, 第 18 卷, 第 5 期, 第 851 页

- ODS 和应用程序 ” , *arXiv preprint arXiv:2302.02591* , 2023 年。5.2 [226] H. Koo 和 T. E. Kim , “ 结构化数据生成扩散模型的全面调查 ” , *arXiv preprint arXiv:2306.04139* , 2023 年。5.2 [227] Y. Song 和 S. Ermon , “ 通过数据分布的梯度进行生成建模 ” , *NeurIPS* , 2019 年。2 , 3 , 5 , 7 [228] A. B orji , “ GaN 评估措施的利与弊：新发展 ” , *Comput Vis Image Underst* , 第 215 卷 , 第 103329 页 , 2022 年。B.1 [229] T. Salimans , I. Goodfellow , W. Zaremba , V. Cheung , A. Radford , and X. Chen , “ 改进的甘斯训练技术 ” , *NeurIPS* , 第29卷 , 2016年。B.1 [230] S. Kullback , *Information theory and statistics*.信使公司 , 1997 年。B.1 [231] A. Razavi、A. Van den Oord 和 O. Vinyals , “ 使用 vq-vae-2 生成多样化的低保真图像 ” , *NeurIPS* , 第 32 卷 , 2019 年。B.3 [232] A. Krizhevsky , G. Hinton *et al.* , “ 从微小的图像中学习多层特征 ” , 2009 年。C [233] A. Krizhevsky、I. Sutskever 和 G. E. Hinton , “ 使用深度卷积神经网络进行 ImageNet 分类 ” , *Communications of the ACM* , 第 60 卷 , 第 6 期 , 第 84-90 页 , 2017 年。
- rocomputing* , 2022.8 [246] A. Lugmayr、M. Danelljan、A. Romero、F. Yu、R. Timofte 和 L. Van Gool , “ 重绘：使用去噪扩散概率模型的修复 ” , *CVPR* , 2022 年 , 第 11 461-11 471 页。8 [247] G. Giannone、D. Nielsen 和 O. Winther , “ 小样本扩散模型 ” , *arXiv:2205.15463* , 2022 年。8 [248] X. Han、H. Zheng 和 M. 周 , “ 卡片：分类和回归扩散模型 ” , *arXiv:2206.07275* , 2022 年。8 [249] T. Amit、E. Nachmani、T. Shaharabany 和 L. Wolf , “ Segdiff：使用扩散概率模型进行图像分割 ” , *arXiv:2112.00390* , 2021 年。8 [250] A.-C.Cheng , X. Li , S. Liu , M. Sun , 和 M.-H.Yang , “ 通过规范映射生成自回归 3D 形状 ” , *arXiv:2204.01955* , 2022 年。8 [251] Y. Song , L. Shen , L. Xing , and S. Ermon , “ Solving inverse problems in medical imaging with score-based generative models ” , *ICLR* , 2021年。8 [252] T. Chen、R. Zhang 和 G. Hinton , “ 模拟比特：使用具有自调节功能的扩散模型生成离散数据 ” , *arXiv:2208.04202* , 2022 年。8 [253] J. M. L. Alcaraz 和 N. Strodthof

附录 A 采样算法

在本节中，我们简要介绍了当前主流采样方法。我们将它们分为两部分：无条件采样和有条件采样。对于无条件采样，我们提出了三个地标的原始采样算法。对于条件采样，我们将它们分为 labeled condition 和 unlabeled condition。

A.1 无条件采样

A.1.1 Ancestral Sampling

Algorithm 1 Ancestral Sampling [10]

```

 $x_T \sim \mathcal{N}(0, I)$ 
for  $t = T, \dots, 1$  do
   $z \sim \mathcal{N}(0, I)$ 
   $x_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left( x_t - \frac{1-\alpha_t}{\sqrt{1-\alpha_t}} \epsilon_\theta(x_t, t) \right) + \sigma_t z$ 
end for
return  $x_0$ 

```

A.1.2 Annealed Langevin Dynamics Sampling

Algorithm 2 Annealed Langevin Dynamics Sampling [227]

```

Initialize  $x_0$ 
for  $i = 1, \dots, L$  do
   $\alpha_i \leftarrow \epsilon \cdot \sigma_i^2 / \sigma_L^2$ 
  for  $t = 1, \dots, L$  do
     $z_t \sim \mathcal{N}(0, I)$ 
     $\tilde{x}_t = \tilde{x}_{t-1} + \frac{\alpha_t}{2} s_\theta(\tilde{x}_{t-1}, \sigma_t) + \sqrt{\alpha_t} z_t$ 
  end for
   $\tilde{x}_0 \leftarrow \tilde{x}_T$ 
end for
return  $\tilde{x}_T$ 

```

A.1.3 Predictor-Corrector Sampling

Algorithm 3 Predictor-Corrector Sampling [11]

```

 $x_N \sim \mathcal{N}(0, \sigma_{\max}^2 I)$ 
for  $i = N - 1$  to  $0$  do
   $z \sim \mathcal{N}(0, I)$ 
  if Variance Exploding SDE then
     $x'_i \leftarrow x_{i+1} + \left( \sigma_{i+1}^2 - \sigma_i^2 \right) s_\theta(x_{i+1}, \sigma_{i+1})$ 
     $x_i \leftarrow x'_i + \sqrt{\sigma_{i+1}^2 - \sigma_i^2} z$ 
  else if Variance Preserving SDE then
     $x'_i \leftarrow \left( 2 - \sqrt{1 - \beta_{i+1}} \right) x_{i+1} + \beta_{i+1} s_\theta(x_{i+1}, i + 1)$ 
     $x_i \leftarrow x'_i + \sqrt{\beta_{i+1}} z$ 
  end if
  for  $j = 1$  to  $M$  do
     $z \sim \mathcal{N}(0, I)$ 
     $x_i \leftarrow x_i + \epsilon_i s_\theta(x_i, \sigma_i) + \sqrt{2\epsilon_i} z$ 
  end for
end for
return  $x_0$ 

```

A.2 条件采样

A.2.1 Labeled Condition

Algorithm 4 Classifier-guided Diffusion Sampling [27]

```

Input: class label  $y$ , gradient scale  $s$ 
 $x_T \sim \mathcal{N}(0, I)$ 
for  $t = T, \dots, 1$  do
  if DDPM Sampling then
     $\mu, \Sigma \leftarrow \mu_\theta(x_t), \Sigma_\theta(x_t)$ 
     $x_{t-q} \leftarrow \text{sample from } \mathcal{N}(\mu + s \Sigma \nabla_{x_t} \log p_\phi(y | x_t), \Sigma)$ 
  end if
  if DDIM Sampling then
     $\hat{\epsilon} \leftarrow \epsilon_\theta(x_t) - \sqrt{1 - \bar{\alpha}_t} \nabla_{x_t} \log p_\phi(y | x_t)$ 
     $x_{t-1} \leftarrow \sqrt{\bar{\alpha}_{t-1}} \left( \frac{x_t - \sqrt{1 - \bar{\alpha}_t} \hat{\epsilon}}{\sqrt{\bar{\alpha}_t}} \right) + \sqrt{1 - \bar{\alpha}_{t-1}} \hat{\epsilon}$ 
  end if
end for
return  $x_0$ 

```

Algorithm 5 Classifier-free Guidance Sampling [48]

```

Input: guidance  $w$ , conditioning  $c$ , SNR  $\lambda_1, \dots, \lambda_T$ 
 $z \sim \mathcal{N}(0, I)$ 
for  $t = 1, \dots, T$  do
   $\tilde{\epsilon}_t = (1 + w) \epsilon_\theta(z_t, c) - w \epsilon_\theta(z_t)$ 
   $\tilde{x}_t = (z_t - \sigma_{\lambda_t} \tilde{\epsilon}_t) / \alpha_{\lambda_t}$ 
   $z_{t+1} \sim \mathcal{N}(\tilde{\mu}_{\lambda_{t+1}|\lambda_t}(z_t, \tilde{x}_t), (\tilde{\sigma}_{\lambda_{t+1}|\lambda_t})^{1-\nu} (\sigma_{\lambda_t|\lambda_{t+1}}^2)^\nu)$ 
end for
return  $z_{T+1}$ 

```

A.3 未标记条件

Algorithm 6 Self-guided Conditional Sampling [56]

```

Input: guidance  $w$ , annotation map  $f_\psi, g_\phi$ , dataset  $\mathcal{D}$ , label  $\mathbf{k}$ , segmentation label  $\mathbf{k}_s$ , image guidance  $\hat{\mathbf{k}}$ 
 $x_T \sim \mathcal{N}(0, I)$ 
for  $t = T, \dots, 1$  do
   $z \sim \mathcal{N}(0, I)$ 
  if Self Guidance then
     $\tilde{\epsilon} \leftarrow (1 - w) \epsilon_\theta(x_t, t) + w \epsilon_\theta(x_t, t; f_\psi(g_\phi(x; \mathcal{D}); \mathcal{D}))$ 
  else if Self-Labeled Guidance then
     $\tilde{\epsilon} \leftarrow \epsilon_\theta(x_t, \text{concat}[t, \mathbf{k}])$ 
  else if Self-Boxed Guidance then
     $\tilde{\epsilon} \leftarrow \epsilon_\theta(\text{concat}[x_t, \mathbf{k}_s], \text{concat}[t, \mathbf{k}])$ 
  else if Self-Segmented Guidance then
     $\tilde{\epsilon} \leftarrow \epsilon_\theta(\text{concat}[x_t, \mathbf{k}_s], \text{concat}[t, \hat{\mathbf{k}}])$ 
  end if
   $x_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left( x_t - \frac{1-\alpha_t}{\sqrt{1-\alpha_t}} \tilde{\epsilon} \right) + \sigma_t z$ 
end for
return  $x_0$ 

```

附录 B 评估指标

B.1 初始分数 (IS)

初始分数建立在对基于 ImageNet 数据集的生成图像的多样性和分辨率的重视之上 [228, 229]。可分为多样性测量和质量测量两部分。用 p_{IS} 表示的多样性度量是根据生成样本的类熵计算的：熵越大，样本就越多样化。 q_{IS} 表示的质量测量是使用熵通过样本和相关类图像之间的相似性来计算的。这是因为如果样本更接近 ImageNet 数据集中的特定图像类别，它们将享有高分辨率。因此，为了降低 q_{IS} 和提高 p_{IS} ，KL 散度 [230] 应用于初始分数计算：

$$\begin{aligned} IS &= D_{KL}(p_{IS} \parallel q_{IS}) \\ &= \mathbb{E}_{x \sim p_{IS}} \left[\log \frac{p_{IS}}{q_{IS}} \right] \\ &= \mathbb{E}_{x \sim p_{IS}} [\log(p_{IS}) - \log(q_{IS})] \end{aligned} \quad (19)$$

B.2 Frechet 初始距离 (FID)

尽管 Inception Score 中有合理的评估技术，但该建立是基于一个具有 1000 个类的特定数据集和一个由随机性（如初始权重和代码框架）组成的训练网络。因此，ImageNet 和真实世界图像之间的偏差可能会导致结果不准确。

FID 的提出是为了解决特定参考数据集的偏差。该分数使用平均值和协方差显示实际数据分布与生成的样本之间的距离。

$$FID = \|\mu_r - \mu_g\|^2 + \text{Tr} \left(\Sigma_r + \Sigma_g - 2(\Sigma_r \Sigma_g)^{1/2} \right) \quad (20)$$

其中 μ_g , Σ_g 是生成样本的均值和协方差， μ_r , Σ_r 是真实数据的均值和协方差。

B.3 负对数似然 (NLL)

根据 Razavi *et al.*, [231] 负对数似然被视为描述所有数据分布模式的常见评估指标。一些扩散模型，如改进的 DDPM [78]，将 NLL 视为分布匹配的度量。

$$NLL = \mathbb{E} [-\log p_{\theta}(x)] \quad (21)$$

附录 C 基准测试

在包括 CIFAR-10 [232]、ImageNet[233] 和 CelebA-64 [234] 在内的各种数据集上提供了地标模型的基准以及与 FID 分数、Inception 分数和 NLL 相对应的改进技术。所选性能根据 NFE 按定义顺序列出，以便于比较。

C.1 CelebA-64 基准测试

表 3 CelebA-64 的基准测试

Method	NFE	FID	NLL
NPR-DDIM [83]	1000	3.15	-
SN-DDIM [83]	1000	2.90	-
NCSN [227]	1000	10.23	-
NCSN ++ [11]	1000	1.92	1.97
DDPM ++ [11]	1000	1.90	2.10
DiffuseVAE [87]	1000	4.76	-
Analytic DPM [82]	1000	-	2.66
ES-DDPM [73]	200	2.55	-
PNDM [46]	200	2.71	-
ES-DDPM [73]	100	3.01	-
PNDM [46]	100	2.81	-
Analytic DPM [82]	100	-	2.66
NPR-DDIM [83]	100	4.27	-
SN-DDIM [83]	100	3.04	-
ES-DDPM [73]	50	3.97	-
PNDM [46]	50	3.34	-
NPR-DDIM [83]	50	6.04	-
SN-DDIM [83]	50	3.83	-
DPM-Solver Discrete [47]	36	2.71	-
ES-DDPM [73]	20	4.90	-
PNDM [46]	20	5.51	-
DPM-Solver Discrete [47]	20	2.82	-
ES-DDPM [73]	10	6.44	-
PNDM [46]	10	7.71	-
Analytic DPM [82]	10	-	2.97
NPR-DDPM [83]	10	28.37	-
SN-DDPM [83]	10	20.60	-
NPR-DDIM [83]	10	14.98	-
SN-DDIM [83]	10	10.20	-
DPM-Solver Discrete [47]	10	6.92	-
ES-DDPM [73]	5	9.15	-
PNDM [46]	5	11.30	-

C.2 ImageNet-64 上的基准测试 C.3 CIFAR-10 数据集上的基准测试

附录 D 改进算法的详细信息 附录 E 符号表

表 4 ImageNet-64 的
基准测试

Method	NFE	FID	IS	NLL
MCG [160]	1000	25.4	-	-
Analytic DPM [82]	1000	-	-	3.61
ES-DDPM [73]	900	2.07	55.29	-
Restart [30]	623	1.36	-	-
Efficient Sampling [85]	256	3.87	-	-
Analytic DPM [82]	200	-	-	3.64
NPR-DDPM [83]	200	16.96	-	-
SN-DDPM [83]	200	16.61	-	-
ES-DDPM [73]	100	3.75	48.63	-
DPM-Solver Discrete [47]	57	17.47	-	-
Restart [30]	39	2.38	-	-
ES-DDPM [73]	25	3.75	48.63	-
GGDM [84]	25	18.4	18.12	-
Analytic DPM [82]	25	-	-	3.83
NPR-DDPM [83]	25	28.27	-	-
SN-DDPM [83]	25	27.58	-	-
DPM-Solver Discrete [47]	20	18.53	-	-
ES-DDPM [73]	10	3.93	48.81	-
GGDM [84]	10	37.32	14.76	-
DPM-Solver Discrete [47]	10	24.4	-	-
ES-DDPM [73]	5	4.25	48.04	-
GGDM [84]	5	55.14	12.9	-

表 5 CIFAR-10 基准 (NF
E ≥ 1000)

Method	NFE	FID	IS	NLL
Improved DDPM [78]	4000	2.90	-	-
VE SDE [11]	2000	2.20	9.89	-
VP SDE [11]	2000	2.41	9.68	3.13
sub-VP SDE [11]	2000	2.41	9.57	2.92
DDPM [10]	1000	3.17	9.46	3.72
NCSN [227]	1000	25.32	8.87	-
SSM [235]	1000	54.33	-	-
NCSNv2 [236]	1000	10.87	8.40	-
D3PM [16]	1000	7.34	8.56	3.44
Efficient Sampling [111]	1000	2.94	-	-
NCSN++ [11]	1000	2.33	10.11	3.04
DDPM++ [11]	1000	2.47	9.78	2.91
TDPM [71]	1000	3.07	9.24	-
VDM [77]	1000	4.00	-	-
DiffuseVAE [87]	1000	8.72	8.63	-
Analytic DPM [82]	1000	-	-	3.59
NPR-DDPM [83]	1000	4.27	-	-
SN-DDPM [83]	1000	4.07	-	-
Gotta Go Fast VP [29]	1000	2.49	-	-
Gotta Go Fast VE [29]	1000	3.14	-	-
INDM [88]	1000	2.28	-	3.09

表 6 CIFAR-10 基准测试
(NFE < 1000)

Method	NFE	FID	IS	NLL
Diffusion Step [75]	600	3.72	-	-
ES-DDPM [73]	600	3.17	-	-
Diffusion Step [75]	400	14.38	-	-
Diffusion Step [75]	200	5.44	-	-
NPR-DDPM [83]	200	4.10	-	-
SN-DDPM [83]	200	3.72	-	-
Gotta Go Fast VP [29]	180	2.44	-	-
Gotta Go Fast VE [29]	180	3.40	-	-
LSGM [35]	138	2.10	-	-
PFGM [32]	110	2.35	-	-
DDIM [26]	100	4.16	-	-
FastDPM [79]	100	2.86	-	-
TDPM [71]	100	3.10	9.34	-
NPR-DDPM [83]	100	4.52	-	-
SN-DDPM [83]	100	3.83	-	-
DiffuseVAE [87]	100	11.71	8.27	-
DiffFlow [237]	100	14.14	-	3.04
Analytic DPM [82]	100	-	-	3.59
Efficient Sampling [111]	64	3.08	-	-
DPM-Solver [47]	51	2.59	-	-
DDIM [26]	50	4.67	-	-
FastDPM [79]	50	3.2	-	-
NPR-DDPM [83]	50	5.31	-	-
SN-DDPM [83]	50	4.17	-	-
Improved DDPM [78]	50	4.99	-	-
TDPM [71]	50	3.3	9.22	-
DEIS [111]	50	2.57	-	-
gDDIM [81]	50	2.28	-	-
DPM-Solver Discrete [47]	44	3.48	-	-
STF [43]	35	1.90	-	-
EDM [25]	35	1.79	-	-
PFGM++ [33]	35	1.74	-	-
Improved DDPM [78]	25	7.53	-	-
GGDM [84]	25	4.25	9.19	-
NPR-DDPM [83]	25	7.99	-	-
SN-DDPM [83]	25	6.05	-	-
DDIM [26]	20	6.84	-	-
FastDPM [79]	20	5.05	-	-
DEIS [111]	20	2.86	-	-
DPM-Solver [47]	20	2.87	-	-
DPM-Solver Discrete [47]	20	3.72	-	-
Efficient Sampling [111]	16	3.41	-	-
NPR-DDPM [83]	10	19.94	-	-
SN-DDPM [83]	10	16.33	-	-
DDIM [26]	10	13.36	-	-
FastDPM [79]	10	9.90	-	-
GGDM [84]	10	8.23	8.90	-
Analytic DPM [82]	10	-	-	4.11
DEIS [111]	10	4.17	-	-
DPM-Solver [47]	10	6.96	-	-
DPM-Solver Discrete [47]	10	10.16	-	-
Progressive Distillation [31]	8	2.57	-	-
Denosing Diffusion GAN [86]	8	4.36	9.43	-
GGDM [84]	5	13.77	8.53	-
DEIS [111]	5	15.37	-	-
Progressive Distillation [31]	4	3.00	-	-
TDPM [71]	4	3.41	9.00	-
Denosing Diffusion GAN [86]	4	3.75	9.63	-
Progressive Distillation [31]	2	4.51	-	-
TDPM [71]	2	4.47	8.97	-
Denosing Diffusion GAN [86]	2	4.08	9.80	-
Denosing student [61]	1	9.36	8.36	-
Progressive Distillation [31]	1	9.12	-	-
TDPM [71]	1	8.91	8.65	-

表 7 改进的扩散方法的
详细信息

Method	Year	Data	Model	Framework	Training	Sampling	Code
Landmark Works							
DPM [9]	2015	RGB Image	Discrete	Diffusion	L_{simple}	Ancestral	[code]
DDPM [10]	2020	RGB Image	Discrete	Diffusion	L_{simple}	Ancestral	[code]
NCSN [227]	2019	RGB Image	Discrete	Score	L_{DSM}	Langevin dynamics	[code]
NCSNv2 [236]	2020	RGB Image	Discrete	Score	L_{DSM}	Langevin dynamics	[code]
Score SDE [11]	2020	RGB Image	Continuous	SDE	L_{DSM}	PC-Sampling	[code]
Improved Works							
Progressive Distill [31]	2022	RGB Image	Discrete	Diffusion	L_{simple}	DDIM Sampling	[code]
Denosing Student [61]	2021	RGB Image	Discrete	Diffusion	$L_{Distill}$	DDIM Sampling	[code]
TDPM [71]	2022	RGB Image	Discrete	Diffusion	$L_{DDPM\&GAN}$	Ancestral	-
ES-DDPM [73]	2022	RGB Image	Discrete	Diffusion	$L_{DDPM\&VAE}$	Conditional Sampling	[code]
CCDF [58]	2021	RGB Image	Discrete	SDE	L_{simple}	Langevin dynamics	[code]
Franzese’s Model [75]	2022	RGB Image	Continuous	SDE	L_{DSM}	DDIM Sampling	-
FastDPM [79]	2021	RGB Image	Discrete	Diffusion	L_{simple}	DDIM Sampling	[code]
Improved DDPM [78]	2021	RGB Image	Discrete	Diffusion	L_{hybrid}	Ancestral	[code]
VDM [77]	2022	RGB Image	Both	Diffusion	L_{simple}	Ancestral	[code]
San-Roman’s Model [80]	2021	RGB Image	Discrete	Diffusion	$L_{DDPM\&Noise}$	Ancestral	-
Analytic-DPM [82]	2022	RGB Image	Discrete	Score	$L_{Trajectory}$	Ancestral	[code]
NPR-DDPM [83]	2022	RGB Image	Discrete	Diffusion	$L_{DDPM\&Noise}$	Ancestral	[code]
SN-DDPM [83]	2022	RGB Image	Discrete	Score	L_{square}	Ancestral	[code]
DDIM [26]	2021	RGB Image	Discrete	Diffusion	L_{simple}	DDIM Sampling	[code]
gDDIM [81]	2022	RGB Image	Continuous	SDE&ODE	L_{DSM}	PC-Sampling	[code]
INDM [88]	2022	RGB Image	Continuous	SDE	$L_{DDPM\&Flow}$	PC-Sampling	-
Gotta Go Fast [29]	2021	RGB Image	Continuous	SDE	L_{DSM}	Improved Euler	[code]
DPM-Solver [47]	2022	RGB Image	Continuous	ODE	L_{DSM}	Higher ODE solvers	[code]
Restart [30]	2023	RGB Image	Continuous	SDE	L_{DSM}	2 nd Order Heun	[code]
EDM [25]	2022	RGB Image	Continuous	ODE	L_{DSM}	2 nd Order Heun	[code]
PFGM [32]	2022	RGB Image	Continuous	ODE	L_{DSM}	ODE-Solver	[code]
PFGM++ [33]	2023	RGB Image	Continuous	ODE	L_{DSM}	2 nd Order Heun	[code]
PNDM [46]	2022	Manifold	Discrete	ODE	L_{simple}	Multi-step & Runge-Kutta	[code]
DDSS [84]	2021	RGB Image	Discrete	Diffusion	L_{simple}	Dynamic Programming	-
GGDM [85]	2022	RGB Image	Discrete	Diffusion	L_{KID}	Dynamic Programming	-
Diffusion GAN [86]	2022	RGB Image	Discrete	Diffusion	$L_{DDPM\&GAN}$	Ancestral	[code]
DiffuseVAE [87]	2022	RGB Image	Discrete	Diffusion	$L_{DDPM\&VAE}$	Ancestral	[code]
DiffFlow [237]	2021	RGB Image	Discrete	SDE	L_{DSM}	Langevin & Flow Sampling	[code]
LSGM [35]	2021	RGB Image	Continuous	ODE	$L_{DDPM\&VAE}$	ODE-Slover	[code]
Score-flow [103]	2021	Dequantization	Continuous	SDE	L_{DSM}	PC-Sampling	[code]
PDM [114]	2022	RGB Image	Continuous	SDE	L_{Gap}	PC-Sampling	-
ScoreEBM [238]	2021	RGB Image	Discrete	Score	$L_{Recovery}$	Langevin dynamics	[code]
Song’s Model [239]	2021	RGB Image	Discrete	Score	L_{DSM}	Langevin dynamics	-
Huang’s Model [104]	2021	RGB Image	Continuous	SDE	L_{DSM}	SDE-Solver	[code]
De Bortoli’s Model [240]	2021	RGB Image	Continuous	SDE	L_{DSM}	Importance Sampling	[code]
PVD [241]	2021	Point Cloud	Discrete	Diffusion	L_{simple}	Ancestral	[code]
Luo’s Model [151]	2021	Point Cloud	Discrete	Diffusion	L_{simple}	Ancestral	[code]
Lyu’s Model [144]	2022	Point Cloud	Discrete	Diffusion	L_{simple}	Farthest Point Sampling	[code]
D3PM [16]	2021	Categorical Data	Discrete	Diffusion	L_{hybrid}	Ancestral	[code]
Argmax [92]	2021	Categorical Data	Discrete	Diffusion	$L_{DDPM\&Flow}$	Gumbel sampling	[code]
ARDM [93]	2022	Categorical Data	Discrete	Diffusion	L_{simple}	Ancestral	[code]
Campbell’s Model [96]	2022	Categorical Data	Continuous	Diffusion	$L_{CT\ simple}$	PC-Sampling	[code]
VQ-diffusion [94]	2022	Vector-Quantized	Discrete	Diffusion	L_{simple}	Ancestral	[code]
Improved VQ-Diff [95]	2022	Vector-Quantized	Discrete	Diffusion	L_{simple}	Purity Prior Sampling	[code]
Cohen’s Model [125]	2022	Vector-Quantized	Discrete	Diffusion	L_{simple}	Ancestral & VAE Sampling	[code]
Xie’s Model [126]	2022	Vector-Quantized	Discrete	Diffusion	$L_{DDPM\&Class}$	Ancestral & VAE Sampling	-
RGSM [97]	2022	Manifold	Continuous	SDE	L_{DSM}	Geodesic Random Walk	-
RDM [98]	2022	Manifold	Continuous	SDE	$L_{CT\ simple}$	Importance Sampling	-
EDP-GNN [22]	2020	Graph	Discrete	Score	L_{DSM}	Langevin dynamics	[code]

表 8 扩散应用的详细
信息

Method	Year	Data	Framework	Downstream Task	Code
Computer Vision					
CMDE [13]	2021	RGB-Image	SDE	Inpainting, Super-Resolution, Edge to image translation	[code]
DDRM [242]	2022	RGB-Image	Diffusion	Super-Resolution, Deblurring, Inpainting, Colorization	[code]
Palette [243]	2022	RGB-Image	Diffusion	Colorization, Inpainting, Uncropping, JPEG Restoration	[code]
DiffC [244]	2022	RGB-Image	SDE	Compression	-
SRDiff [245]	2021	RGB-Image	Diffusion	Super-Resolution -	-
RePaint [246]	2022	RGB-Image	Diffusion	Inpainting, Super-resolution, Edge to Image Translation	[code]
FSDM [247]	2022	RGB-Image	Diffusion	Few-shot Generation	-
CARD [248]	2022	RGB-Image	Diffusion	Conditional Generation	[code]
GLIDE [49]	2022	RGB-Image	Diffusion	Conditional Generation	[code]
LSGM [35]	2022	RGB-Image	SDE	UnConditional & Conditional Generation	[code]
SegDiff [249]	2022	RGB-Image	Diffusion	Segmentation	-
VQ-Diffusion [94]	2022	VQ Data	Diffusion	Text-to-Image Synthesis	[code]
DreamFusion [152]	2023	VQ Data	Diffusion	Text-to-Image Synthesis	[code]
Text-to-Sign VQ [126]	2022	VQ Data	Diffusion	Conditional Pose Generation	-
Improved VQ-Diff [95]	2022	VQ Data	Diffusion	Text-to-Image Synthesis	-
Luo's Model [151]	2021	Point Cloud	Diffusion	Point Cloud Generation	[code]
PVD [147]	2022	Point Cloud	Diffusion	Point Cloud Generation, Point-Voxel representation	[code]
PDR [144]	2022	Point Cloud	Diffusion	Point Cloud Completion	[code]
Cheng's Model [250]	2022	Point Cloud	Diffusion	Point Cloud Generation	[code]
Luo's Model [150]	2022	Point Cloud	Score	Point Cloud Denoising	[code]
VDM [12]	2022	Video	Diffusion	Text-Conditioned Video Generation	[code]
RVD [157]	2022	Video	Diffusion	Video Forecasting, Video compression	[code]
FDM [156]	2022	Video	Diffusion	Video Forecasting, Long-range Video modeling	-
MCVD [155]	2022	Video	Diffusion	Video Prediction, Video Generation, Video Interpolation	[code]
RaMViD [158]	2022	Video	SDE	Conditional Generation	-
Score-MRI [161]	2022	MRI	SDE	MRI Reconstruction	[code]
Song's Model [251]	2022	MRI, CT	SDE	MRI Reconstruction, CT Reconstruction	[code]
R2D2+ [167]	2022	MRI	SDE	MRI Denoising	-
Sequence Modeling					
Diffusion-LM [15]	2022	Text	Diffusion	Conditional Text Generation	[code]
Bit Diffusion [252]	2022	Text	Diffusion	Image-Conditional Text Generation	[code]
D3PM [16]	2021	Text	Diffusion	Text Generation	-
Argmax [92]	2021	Text	Diffusion	Test Segmentation, Text Generation	[code]
CSDI [17]	2021	Time Series	Diffusion	Series Imputation	[code]
SSSD [253]	2022	Time Series	Diffusion	Series Imputation	[code]
CSDE [254]	2022	Time Series	SDE	Series Imputation, Series Prediction	-
Audio & Speech					
WaveGrad [19]	2020	Audio	Diffusion	Conditional Wave Generation	[code]
DiffWave [186]	2021	Audio	Diffusion	Conditional & Unconditional Wave Generation	[code]
GradTTS [20]	2021	Audio	SDE	Wave Generation	[code]
Diff-TTS [193]	2021	Audio	Diffusion	non-AR mel-Spectrogram Generation, Speech Synthesis	-
DiffVC [215]	2022	Audio	SDE	Voice conversion	[code]
DiffSVC [190]	2022	Audio	Diffusion	Voice Conversion	[code]
DiffSinger [187]	2022	Audio	Diffusion	Singing Voice Synthesis	[code]
DiffSound [197]	2021	Audio	Diffusion	Text-to-sound Generation tasks	[code]
EdiTTS [192]	2022	Audio	SDE	fine-grained pitch, content editing	[code]
Guided-TTS [196]	2022	Audio	SDE	Conditional Speech Generation	-
Guided-TTS2 [195]	2022	Audio	SDE	Conditional Speech Generation	-
Levkovitch's Model [255]	2022	Audio	SDE	Spectrograms-Voice Generation	[code]
SpecGrad [194]	2022	Audio	Diffusion	Spectrograms-Voice Generation	[code]
ItoTTS [191]	2022	Audio	SDE	Spectrograms-Voice Generation	-
ProDiff [188]	2022	Audio	Diffusion	Text-to-Speech Synthesis	[code]
BinauralGrad [189]	2022	Audio	Diffusion	Binaural Audio Synthesis	-
AI For Science					
ConfGF [256]	2021	Molecular	Score	Conformation Generation	[code]
DGSM [257]	2022	Molecular	Score	Conformation Generation, Sidechain Generation	-
GeoDiff [23]	2022	Molecular	Diffusion	Conformation Generation	[code]
EDM [198]	2022	Molecular	SDE	Conformation Generation	[code]
Torsional Diff [200]	2022	Molecular	Diffusion	Molecular Generation	[code]
DiffDock [203]	2022	Molecular&protein	Diffusion	Conformation Generation, molecular docking	[code]
CDVAE [258]	2022	Protein	Score	Periodic Material Generation	[code]
Luo's Model [24]	2022	Protein	Diffusion	CDR Generation	-
Anand's Model [135]	2022	Protein	Diffusion	Protein Sequence and Structure Generation	-
ProteinSGM [199]	2022	Protein	SDE	de novo protein design	-
DiffFolding [202]	2022	Protein	Diffusion	Protein Inverse Folding	[code]

表 9 扩散系统中的
概念

Notations	Descriptions
T	Discrete total time steps
t	Random time t
z_t	Random noise with normal distribution
ϵ	Random noise with normal distribution
$\mathcal{N}$	Normal distribution
β	Generalized process noise scale
β_t	Variance scale coefficients
$\beta(t)$	Continuous-time β_t
σ	Generalized process noise scale
σ_t	Noise scale of perturbation
$\sigma(t)$	Continuous-time σ_t
α_t	Mean coefficient defined as $1 - \beta_t$
$\alpha(t)$	Continuous-time α_t
$\tilde{\alpha}_t$	Cumulative product of α_t
$\gamma(t)$	Signal-to-Noise ratio
η_t	Step size of annealed Langevin dynamics
x	Unperturbed data distribution
$\tilde{x}$	Perturbed data distribution
x_0	Starting distribution of data
x_t	Diffused data at time t
x'_t	Partly diffused data at time t
x_T	Random noise after diffusion
$F(x, \sigma)$	Forward/Diffusion process
$R(x, \sigma)$	Reverse/Denoised process
$F_t(x_t, \sigma_t)$	Forward/Diffusion step at time t
$R_t(x_t, \sigma_t)$	Reverse/Denoised step at time t
$q(x_t x_{t-1})$	DDPM forward step at time t
$p(x_{t-1} x_t)$	DDPM reverse step at time t
$f(x, t)$	Drift coefficient of SDE
$g(t)$	Simplified diffusion coefficient of SDE
$\mathcal{D}(x, t)$	Degrader at time t in Cold Diffusion
$\mathcal{R}(x, t)$	Reconstructor at time t in Cold Diffusion
$w, \tilde{w}$	Standard Wiener process
$\nabla_x \log p_t(x)$	Score function w.r.t x
$\mu_\theta(x_t, t)$	Mean coefficient of reversed step
$\Sigma_\theta(x_t, t)$	Variance coefficient of reversed step
$\epsilon_\theta(x_t, t)$	Noise prediction model
$s_\theta(x)$	Score network model
L_0, L_{t-1}, L_T	Forward loss, reversed loss, decoder loss
L_{vlb}	Evidence Lower Bound
L_{vlb}^{CT}	Continuous evidence lower bound
L_{simple}	Simplified denoised diffusion loss
L_{simple}^{CT}	Continuous L_{simple}
L_{Gap}	Variational gap
L_{KID}	Kernel inception distance
$L_{Recovery}$	Recovery likelihood loss
L_{hybrid}	Hybrid diffusion loss
$L_{DDPM\&GAN}$	DPM ELBO and GAN hybrid loss
$L_{DDPM\&VAE}$	DPM ELBO and VAE hybrid loss
$L_{DDPM\&Flow}$	DPM ELBO and normalizing flow hybrid loss
L_{DSM}	Loss of denoised score matching
L_{ISM}	Loss of implicit score matching
L_{SSM}	Loss of sliced score matching
$L_{Distill}$	Diffusion distillation loss
$L_{DDPM\&Noise}$	DPM ELBO and reverse noise hybrid loss
L_{Square}	Noise square loss
$L_{Trajectory}$	Process optimization loss
$L_{DDPM\&Class}$	DPM ELBO and classification hybrid loss
θ	learnable parameters
ϕ	learnable parameters