

利用判别性码本先验进行自回归图像生成

Longxiang Tang, Ruihang Chu, Xiang Wang, Yujin Han, Pingyu Wu, Chunming He,
Yingya Zhang, Shiwei Zhang, Jiaya Jia, *Fellow, IEEE*

Abstract—高级离散令牌自回归图像生成系统首先使用码书将图像标记化为令牌索引序列，然后在自回归范式中对这些序列进行建模。尽管自回归生成模型仅在索引值上进行训练，但在码书中编码的先验信息，包括丰富的令牌相似性信息，未被加以利用。最近的研究尝试通过对令牌进行简单的 k-means 聚类来结合这种先验，从而在一个较小的码书内促进生成模型的训练。然而，我们揭示了 k-means 聚类在码书特征空间中表现不佳的问题，这个问题源于固有缺陷，包括令牌空间差异和质心距离不准确。在这项工作中，我们提出了判别性码书先验提取器 (DCPE)，作为 k-means 聚类的替代方案，以更有效地挖掘和利用嵌入在码书中的令牌相似性信息。DCPE 替换了常用的基于质心的距离，这被发现对令牌特征空间不合适且不准确，使用了一个更合理的基于实例的距离。通过使用聚合合并技术，它进一步通过避免分割高密度区域和聚集低密度区域来解决令牌空间差异问题。大量实验表明，DCPE 是即插即用的，并与现有的基于码书先验的范式无缝集成。在提取到判别性先验后，DCPE 加速了自回归模型的训练，在 LlamaGen-B 上提升了 42%，并改善了最终的 FID 和 IS 性能。

Index Terms—Generative Models, Autoregressive Models, Image Generation, Discrete Token, Codebook Prior

I. 介绍

自回归的图像生成 [1], [2], [3], [4], [5], [6] 作为一种新的图像生成范式发展迅速，与传统的扩散方法相比 [7], [8], [9], [10]，显示出越来越有前途的性能和像大型语言模型 (LLMs) [11], [12], [13], [14], [15], [16], [17], [18] 一样扩展的能力。其中，使用离散分词器的方法 [19], [20], [21], [22] 因其与 LLM 对齐的结构 [23], [24] 而受到越来越多的关注，从而实现统一的世界模型。如图 1 (a) 所示，基于标准离散分词自动回归图像生成框架由两个主要部分组成：图像分词器 [25] 和自动回归转换器 [26], [27]。图像分词器由图像编码器、可学习的代码书和图像解码器组成。训练图像首先被输入到编码器中以提取视觉特征，随后通过向量量化将连续特征转换为索引值序列。这种量化是通过识别代码书中与每个视觉特征最接近的分词向量实现的。然后，这些索引序列用于以因果建模方式训练自动回归转换器模型。为了在推理过程中生成新图像，经过良好训练的自动回归模型生成适当的分词索引序列，随后由解码器解码成图像。

大多数现有工作采用的这种自回归图像生成框架 [28], [29], [30], [31]，将图像标记器与自回归模型的训练分离，期

待后者能直接从索引序列中拟合图像标记的分布。然而，由于图像标记器被用于编码和解码，它自然包含了可能有助于生成模型训练的丰富信息。例如，在推理过程中，具有类似语义的标记理想情况下应该共享相似的 logits。一些最近的工作 [32], [33] 开始通过在标记上加入 k-means 聚类操作，研究编码在码本中的标记相似性先验，以促进自回归模型的训练。例如，IAR [34] 鼓励模型通过附加的面向聚类的损失正确预测目标标记所在的聚类，而 CTF [35] 则引入了一个粗到细的管道，通过标记聚类，首先预测聚类索引，然后将其细化为细粒度标记。

虽然取得了一些成功，但我们认为用于提取码本先验的简单 k-means 聚类算法在令牌特征空间中可能表现不佳。为了证明这一点，我们进行了一次简单的实验，如图 1 (b) 所示。我们将来自 LlamaGen [36] 码本的 16k 令牌聚为 8k 个簇。然后使用聚类结果将训练序列中的令牌索引转化为簇索引。在用这些簇索引训练自回归模型之后，我们在预测簇中随机选择令牌以实现适当的图像解码。其背后的直觉是，有效的聚类算法应该能够将具有相似语义的令牌分组到同一簇中。因此，模型应该学会预测准确的簇并保持合理的生成质量。然而，结果显示，模型难以拟合由简单 k-means 聚类生成的簇索引序列。

这些结果表明，简单的 k-means 聚类在捕捉码本中固有的标记相似信息方面表现出了次优的能力。如图 1 (c) 所示，我们将这一现象归因于两个主要因素：(1) 标记空间差异性。自编码器必须在多个粒度上处理信息以压缩和重构图像；因此，标记特征空间在密度上变得不均匀 [37], [38], [39]。由于 k-means 随机初始化聚类，并在更新时不考虑密度变化，它可能会无意中分割高密度区域，并将实际上不相似的稀疏区域的标记组合在一起。(2) 中心点距离不准确。尽管标记特征空间可以看作是高维空间中的流形 [40], [41], [42]，但 k-means 使用向量的算术平均值作为聚类中心点来计算距离的方法并不理想。由于这种有缺陷的距离测量方法，可能导致不相似的标记显得接近。这两个因素导致了聚类内的不相似性，处理这种不准确的聚类结果可能导致次优的生成性能。我们在 Section III-A 中提供了更多的讨论，并在 Section IV-C 中进行了实验演示。

在本文中，我们介绍了判别性码书先验提取器 (DCPE)，旨在更有效地利用嵌入在码书中的标记相似性信息。为了应对标记空间差异问题，我们采用了一个聚合聚类策略，迭代地合并两个最相似的聚类，证明在非均匀标记特征空间中非常有效。这种密度感知策略成功地避免了高密度区域的分裂和低密度区域的聚合。为了解决质心距离不准确的问题，DCPE 使用基于实例的距离测量计算簇间距离，消除了对不准确高维质心的依赖。使用 DCPE，我们能够以聚类码书进行自回归模型的训练，实现加速训练并获得可比甚至更好的性能。例如，在 LlamaGen-B 训练中，加速达到了 42% 对于 FID 和 55% 对于 IS。作为一个即插即用的解决方案，DCPE 还可以无缝集成到现有的基于码书先验的方法中，提供增强的生成性能。

我们的主要贡献可概括为以下三个方面：

Corresponding author: Ruihang Chu.

Longxiang Tang and Jiaya Jia are with the Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong (e-mail: lloong.x@gmail.com; jia@cse.ust.hk).

Ruihang Chu, Xiang Wang, Yujin Han, Pingyu Wu, Yingya Zhang and Shiwei Zhang are with the Tongyi Lab, Alibaba Group, Hangzhou 311121, China (e-mail: ruihangchu@gmail.com).

Chunming He is with the Department of Biomedical Engineering, Duke University, Durham, NC 27708 USA.

This work has been submitted to the IEEE for possible publication. Copyright may be transferred without notice, after which this version may no longer be accessible.

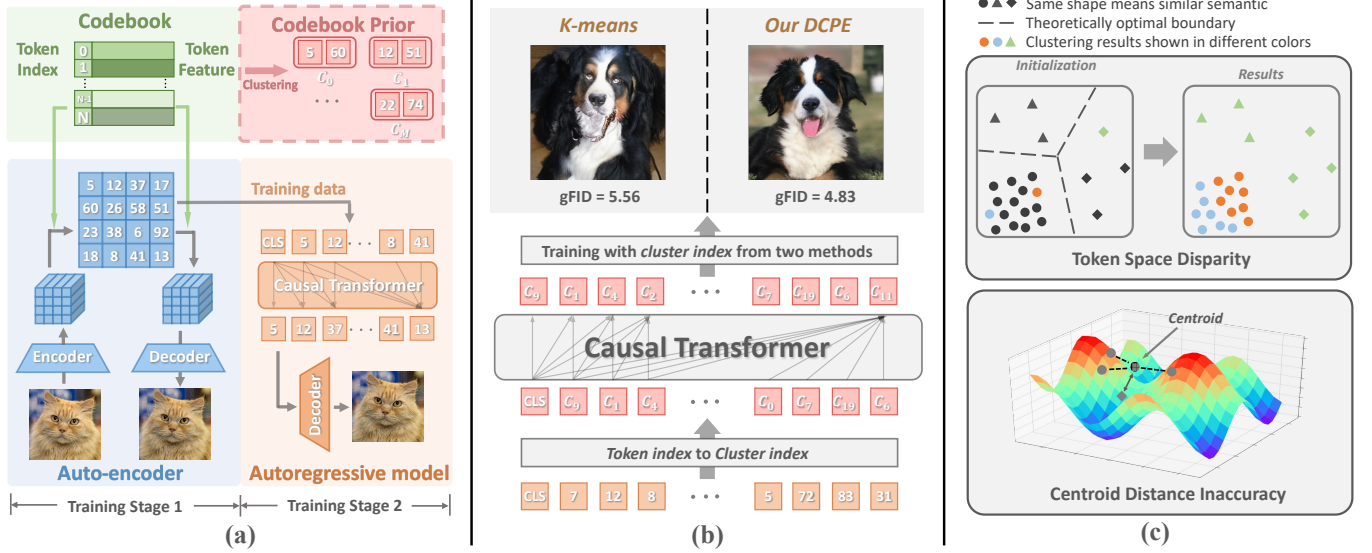


Fig. 1. (a): 一种用于基于离散标记的自回归图像生成方法的通用框架。在分别训练图像标记器和自回归模型时，我们提取标记相似性信息作为码本先验，以辅助后者的训练。(b): 一个关于使用由不同聚类算法产生的聚类索引训练自回归模型的简单实验。K-means 导致性能下降，而我们的 DCPE 实现了更好的生成质量。(c): 标记空间差异和中心点距离不准确问题的示意图，突出了将简单的 k-means 聚类应用于码本标记特征空间的局限性。

- 我们研究并揭示了在为自回归图像生成提取码本先验时的两个关键问题：令牌空间差异和中心距离不准确。
- 我们引入了用于码本先验提取的 DCPE，以替代广泛采用但存在缺陷的 k-means 方法，通过聚合策略和基于实例的距离计算来解决已识别的问题。
- 实验结果表明，我们的方法是即插即用的，并且可以无缝集成到现有工作中，在减少码本大小的情况下提供了更好的性能。

II. 相关工作

自回归图像生成。随着扩散模型 [1], [2], [3], [4], [5], [6], [7] 和自回归范式 [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18], [19], [20], [21], [22], [23], [24], [25], [26], [27], [28], [29], [30], [31], [32], [33], [34], [35], [36], [37], [38], [39], [40], [41], [42], [43], [44], [45], [46], [47], [48], [49], [50], [51], [52], [53], [54], [55], [56], [57], [58], [59], [60], [61], [62], [63], [64], [65], [66], [67], [68], [69], [70], [71], [72], [73], [74], [75], [76], [77], [78], [79], [80], [81], [82], [83], [84], [85], [86], [87], [88], [89], [90], [91], [92], [93], [94], [95], [96], [97], [98], [99], [100] 等生成视觉架构的快速发展，视觉生成取得了显著进展，并能够合成高保真度的图像内容。其中，自回归范式通过基于先前的视觉内容预测序列中的下一个视觉内容，最近受到了极大关注，这一灵感来源于自然语言处理领域的巨大成功 [1], [2], [3], [4]。LlamaGen [5] 适应了基于 Llama [6] 架构的“下一个标记预测”方式，尝试自回归地预测下一个视觉标记。MAR [7] 利用扩散头来建模每个标记的概率，消除了对离散分词器的需求。VAR [8] 引入了下一个尺度预测范式，以由粗到细的分辨率预测形式生成视觉内容。TiTok [9] 探索了一种基于一维 Transformer 的分词器，通过插入额外的可学习标记将二维图像转换为一维潜在序列。RAR [10] 将光栅顺序标记预测置换为不同的分解顺序，增强了捕捉双向依赖性的能力。在这项工作中，我们专注于基于离散标记的自回归图像生成，旨在结合编码在代码簿中的先验以增强生成能力。

码本操控。虽然使用离散编码器的图像生成方法吸引了越来越多的注意力，但一些研究已经开始探索操控码本的方法，以改善或加速生成模型的训练。HyperHill [11] 通过共现感知超球面嵌入增强码本向量，并通过 Hilbert 曲线重新排列它们，以实现码本的灵活调整，提高效率 and 重构质量。CVQ-VAE [12] 通过使用编码特征作为锚点来更新

不活动的码向量，从而缓解码本塌陷，提高码本利用率与现有架构的集成。RAQ [13] 通过以数据驱动的方式调整码本，使基于 VQ 的生成模型支持多个比特率，且无需重新训练。虽然这些方法都专注于调整码本，但它们并没有将任何码本信息纳入生成模型的训练过程中。相比之下，我们的工作旨在从码本中提取先验知识用于生成模型的训练，提供与现有方法的即插即用集成。

使用码书先验进行训练。现有的离散编码器自回归图像生成框架通常将图像编码器和生成模型的训练分开。最近的研究 [14], [15], [16] 开始探索如何利用从码书中获得的令牌相似性信息来提高生成质量。IAR [17] 提出了采用均衡 k 均值聚类的码书重排策略，并使用一个面向聚类的交叉熵损失来指导模型正确预测目标令牌所在的聚类，从而提高生成质量和鲁棒性。CTF [18] 引入了一种粗到细的管道，训练一个自回归模型按顺序预测通过聚类获得的粗略标签，以及一个辅助模型在粗略标签的条件下预测细粒度标签。RobustTok [19] 提出通过扰动令牌来训练编码器，这提高了编码器的鲁棒性，从而减轻了重建和生成质量的差异。这些方法通常使用朴素的 k 均值聚类或 top-k 选择提取码书先验，而不分析令牌特征分布的底层属性。在本研究中，我们通过提出一个更适合令牌特征空间的提取器来解决这一差距，从而利用更具辨别力的码书先验。

III. 方法

A. 预备知识

我们首先描述大多数现有的基于离散标记的自回归图像生成方法的训练和推理过程。给定一个由编码器、解码器和码本组成的图像分词器，我们可以将图像转化为一个索引序列，用于训练自回归模型。首先，图像特征由编码器提取，其中代表每个特征向量的维度。然后，我们应用向量量化对图像特征进行码本化，其中是第个标记向量，并且是码本的大小。具体而言，对于图像特征中的每个特征向量，我们寻找其在码本中最近邻标记的索引：其中是码本中的第个向量，而表示常用于的欧几里得距离。在从所

有训练图像中获得量化索引序列之后，我们可以使用它们通过下一个标记预测范式来训练一个自回归生成模型。在推理过程中，训练良好的自回归模型以因果的方式生成一个标记序列，然后通过解码器将其解码为一个图像。

1) K 均值码本聚类：在这里，我们将广泛使用的 k -means 聚类算法应用于代码簿的标记。代码簿聚类的目标是给代码簿中的每个标记 v_i 分配聚类索引 $\{c_i\}_{i=1}^k$ ，其中 $k < N$ 是聚类的数量。 k -means 算法维护一个质心集 $\{\mu_i\}_{i=1}^k$ 并将每个标记分配到最近质心的聚类。首先，质心 $\{\mu_i^{(0)}\}_{i=1}^k$ 通过在代码簿中随机选择的 k 个标记向量进行初始化，初始的聚类分配 $\{c_i^{(0)}\}_{i=1}^k$ 可以通过以下方式获得： $c_i^{(0)} = \arg \min_{j \in [0, k)} \|\mu_j^{(0)} - v_i\|$ ，其中括号中的上标表示不同的迭代。在当前迭代中将聚类索引分配给每个标记后， k -means 通过计算同一聚类中所有标记的算术平均值来更新质心，表示为 $\mu_i^{(1)} = \frac{1}{|C_i^{(0)}|} \sum_{v_j \in C_i^{(0)}} v_j$ ，其中 $C_i^{(0)} = \{v_j | c_j^{(0)} = i\}$ 。这种更新聚类索引和质心的过程不断迭代进行，直到收敛。

尽管 k -means 算法被广泛使用，但在这里的标记特征空间中并不适合用于聚类。这种限制主要来源于高维空间中标记向量的稀疏分布。 k -means 算法可能会将语义不同的标记向量分组到同一簇中，从而导致簇内一致性不佳 [?], [?], [?]。我们将这个问题归因于我们的情境中的两个主要因素：标记空间差异和质心距离不准确。首先，由于码本在多种粒度上编码信息，标记特征空间表现出非均匀密度 [?], [?], [?], [?], [?]。 k -means 算法没有考虑这种变化，同时更新所有簇，这可能导致密集标记区域被分割，而稀疏区域中语义独特的标记被聚在一起。而且，由于标记向量位于高维空间的复杂流形上，它们的算术平均，即簇质心，通常位于流形之外 [?], [?], [?]。这种不对齐导致质心与标记向量之间的距离计算不准确。这些挑战强调了需要更适合于码本标记结构的聚类方法。

B. 判别式码书先验提取器

上述 k -means 聚类中的标记空间差异和中心距离不准确问题可能导致标记聚类包含混合的语义信息，从而为后续自回归模型 [?], [?] 的训练提供不准确的码本先验。为了解决这些挑战，我们提出了判别性码本先验提取器 (DCPE)，以更好地利用嵌入在码本内的标记相似性信息。

k -means 不能解决标记特征空间差异问题的主要原因是它在没有优先级的情况下同时更新所有簇。这种方法不能保证高密度标记的区域优先于低密度区域合并。此外，由于容易出错的随机初始化策略，该问题进一步加剧。为了将高密度标记被分离和低密度区域被合并的风险降到最低，我们借鉴了凝聚聚类的概念来设计一种确定性的聚类算法。我们的方法优先合并两个最相似的簇，确保聚类结果对标记密度敏感。具体来说，我们将所有标记向量 $\{v_i\}_{i=1}^N$ 初始化为单独的簇，即 $C_i^{(0)} = \{v_i\}$ ，然后执行 $N - k$ 次合并操作，以获得 k 个簇。每次，我们贪心地合并两个最相似的标记簇，其表达式如下：

$$C_s^{(i+1)} = C_s^{(i)} \cup C_t^{(i)}, \text{ where } s, t = \arg \min \mathcal{D}(C_s^{(i)}, C_t^{(i)}) \quad (1)$$

其中 $i = 1, 2, \dots, N - k$ 表示迭代次数， $\mathcal{D}(\cdot)$ 是两个簇之间的距离度量。这一策略从设计上确保了高密度区域的簇在低密度区域的簇之前合并，有效防止了包含语义分歧标记的簇的出现，这会妨碍自回归模型的训练。同时，与启

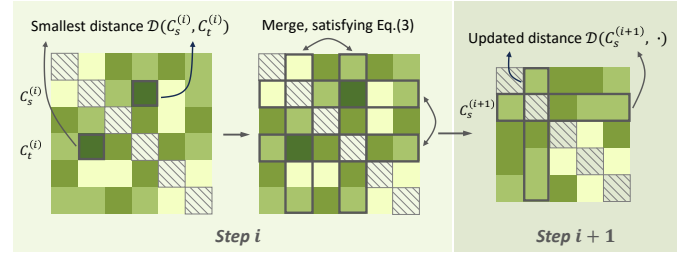


Fig. 2. 说明我们的 DCPE 实现过程。每个网格代表两个簇之间的距离（颜色越深表示距离越小）。我们保持簇间距离，并在层次聚类过程更新距离矩阵，从而加速计算。

发式算法 k -means 相比，这一设计需要更多的计算，后面的小节将讨论这一点。

考虑到关于质心距离不准确问题的讨论，使用质心距离来测量标记特征空间中的簇间相似性是不合适的。因此，我们决定放弃使用质心，而仅依赖有效的标记间距离来间接测量簇间距离。在这方面，DCPE 采纳了一种简单而有效的基于实例的簇间距离方法。对于两个簇 $C_s = \{v_i\}_{i=0}^{n_s}$ 和 $C_t = \{v_j\}_{j=0}^{n_t}$ ，其中 n_s 和 n_t 表示每个簇中标记的数量，我们首先计算两个簇中标记的所有成对距离： $\hat{\mathcal{D}}(v_i, v_j) = \|v_i - v_j\|$ 。然后通过平均所有这些实例级距离来获得簇间距离，其表达式为：

$$\mathcal{D}(C_s, C_t) = \mathbb{E}_{v_i \sim C_s, v_j \sim C_t} [\hat{\mathcal{D}}(v_i, v_j)] \quad (2)$$

此距离在 Eq. (1) 中用于替换现有方法中广泛使用的质心距离。该方法与无权平均连接聚合聚类共享相似的基础理念 [?]，并在 token 特征空间中表现有效，提高了聚类的区分性，从而增强随后的生成模型训练，获得更好的代码簿先验。

C. 实现与分析

根据 Eq. (2)，我们的 DCPE 中计算簇间距离需要计算两个簇中每个 token 间的成对距离。如果算法按照这个公式直接实现，计算复杂度会变得相当高，达到 $O(N^3 d)$ (证明附在附录中)。此外，由于频繁涉及张量切片，该方法未能充分利用 GPU 加速。这个问题在代码本规模较大的情况下尤其严重 [?], [?]。

为了提高我们 DCPE 的效率，从而增强其实用性，我们探索了实现层面的优化。鉴于 DCPE 采用凝聚聚类策略，很容易观察到，未参与合并过程的簇之间的距离无需重新计算；它们可以直接继承自上一轮迭代。然而，对于合并产生的新簇，计算它们与所有剩余簇之间的距离仍然涉及大量的张量切片操作，导致计算效率低下。幸运的是，我们采用的基于实例的簇距离度量可以解释为多个词元之间距离的组合。这使得我们可以将新的距离与现有的距离相结合：通过取两个原簇与其他簇之间距离的平均值实现。通过此优化，我们只需维护一个簇间距离矩阵，该矩阵通过成对的词元距离初始化，并在整个聚类过程中逐步更新，从而减轻冗余计算的负担。其实现示例见 Fig. 2，详细的算法步骤见 Algorithm 1。通过维护和更新簇间距离矩阵，我们显著提高了 DCPE 的效率。如伪代码所示，优化后算法的复杂度降低为 $O(N^3)$ 。这不仅降低了理论复杂度，还在时间复杂度的常数因子上带来显著改善。此外，优化后的实现非常适合并行计算，能够充分利用 GPU 加速。实验结果表明，对于 Llamagen [?] 中码本大小为 16k

的情况，我们的优化实现将处理时间从朴素实现中的 4.4 小时降低到仅 39.6 秒。与模型训练所需的数百个 GPU 小时相比，此优化使得我们的方法在计算成本上可以忽略不计，特别是考虑到这只是生成模型训练前的一次性预处理步骤。

作为编码书令牌特征空间中 k-means 聚类算法的有效替代方案，我们提出的 DCPE 本质上是即插即用的，可以无缝集成到现有基于编码书先验的方法中。为了展示我们 DCPE 的有效性，我们在实验中将其集成到 CTF [?] 和 IAR [?] 的流程中。首先，CTF 训练一个自回归模型，使用粗略的簇索引生成簇索引序列，以及一个细化模型将这些索引转换为实际令牌。我们观察到，有限数量的簇（设置为总令牌数量的 1/32）以及使用次优的 k-means 算法导致了具有多样语义令牌的簇。这需要一个大型细化模型来实现图像生成。我们建议，通过使用相对较多数量的簇的 DCPE，细化模型甚至可以被消除，同时仍然保持竞争力的结果。相应的实验结果显示在 Table I。

IAR 在基本的自回归下一步交叉熵损失上引入了一个辅助的聚类导向损失，通过预测正确的聚类索引来加速训练。在这个框架中整合 DCPE 是直接的。此外，IAR 使用平衡的 k-means 聚类来确保聚类大小相等，允许直接比较不同聚类的 logits，这些 logits 被定义为一个聚类中所有 token logits 的总和。然而，正如前面提到的关于 token 空间差异的问题，这种方法与 token 的实际语义分布不符，并且 DCPE 在其设计中并未对聚类大小施加这样的限制。我们在 Fig. 4 中研究了这种差异。为了使 IAR 中的聚类 logits 与 DCPE 兼容，我们将聚类 logits 重新定义为该聚类中所有 token 的 logits 的平均值。实验结果可以在 Table IV 中找到。

IV. 实验

A. 实现细节

参考现有的基于代码簿的方法 [?], [?]，我们采用 LlamaGen [?] 作为我们的主干网络，并利用其现成的图像 tokenizer。这个 tokenizer 的代码簿大小为 16,384，并以 16 的比例下采样输入图像。对于自回归模型的训练，我们严格遵循 LlamaGen 的原始设置。为了训练具有聚类代码簿的模型，我们根据聚类结果将从训练数据集中提取的 token 索引转换为相应的聚类索引。由于以这种方式训练的生成模型只能预测聚类索引，我们采用一种简单的随机选择策略来解码图像，随机从预测的聚类中选择一个 token。除非另有说明，标记为“+ DCPE”的实验遵循这种解码策略。

为了训练优化模型，我们遵循 vanilla LlamaGen 的训练配置，唯一的修改是将训练周期减少到 100。在模型结构方面，我们基于 LlamaGen-B 代码库，移除因果掩码，这与 CTF [?] 描述的方法一致。此外，如 Table VIII 所示，由于改良的代码簿先验，我们将模型层数减少到一个。在此配置下，我们优化模型的训练时间约为 CTF 中优化模型的 10%，显著提高了训练效率。

对于将 DCPE 集成到 IAR 的实验，我们将簇的数量设置为 512，簇损失权重设置为 1.0，保持所有其他训练设置与我们复现的代码中使用的原始 IAR 一致。此外，由于 Section III-C 中讨论的均值操作，不同簇的 token 梯度不平衡。这种梯度不平衡可能导致簇大小高度偏斜时性能下降。为了缓解这一问题，我们简单地阻止那些大小超过 N/k 的簇合并。虽然这只是一个临时解决方案，但对此问

Algorithm 1 我们的 DCPE 的 PyTorch 伪代码

```
# codebook: size [N, d]
# k: target number of clusters
dist = torch.cdist(codebook, codebook, p=2)
dist.fill_diagonal_(float('inf'))
sizes = torch.ones(N)
labels = torch.arange(N)
for m in range(N, k, -1):
    # get position of nearest clusters
    min_pos = torch.argmax(dist.view(-1), dim=0)
    min_pos_i, min_pos_j = min_pos//m, min_pos % m
    # get real cluster index since dist is merged
    label_i = get_real_label(min_pos_i)
    label_j = get_real_label(min_pos_j)
    # update cluster index label
    labels[labels == label_j] = label_i
    # add j-th row/column to i-th row/column
    sizes = add_j_to_i(sizes, min_pos_i, min_pos_j)
    dist = add_j_to_i(dist, min_pos_i, min_pos_j)
    # remove j-th row and column
    sizes = remove_j(sizes, min_pos_j)
    dist = remove_j(dist, min_pos_j)
```

题的深入研究留待未来工作中进行。更详细的超参数设置可以在附录中找到。

1) 评估：根据 LlamaGen [?]，我们在类条件图像生成任务上使用 ImageNet-1K 基准评估图像生成性能 [?]。生成的图像分辨率为 256×256 ，对应的标记序列长度为 256。对于每次评估，我们在所有 1,000 个类别中总共生成 50,000 张图像。我们使用与现有研究相同的代码计算 FID、初始评分 (IS)、精度 (Pr.) 和召回率 (Re.) [?]。rFID、PSNR 和 SSIM 也用于评估重建性能。FID 衡量训练图像特征与生成图像特征之间的分布相似性；较低的 FID 表示更好的生成质量。IS 通过计算从生成图像中提取的特征的熵来评估图像质量和多样性；较高的 IS 反映更好的质量和多样性。精度和召回率评估类条件生成性能，较高的值表示更高的准确性。rFID 比较重建图像与原始图像，较低的值也表明重建更好。PSNR 量化两个图像之间的像素级差异，而 SSIM 则评估感知相似性。较高的 PSNR 和 SSIM 分数表示重建与原始图像之间有更好的感知对齐。有关生成超参数的更多详细信息可以在附录中找到。

B. 主要结果

正如在 Section III-C 中所讨论的，我们使用从不同聚类算法获得的聚类索引来训练自回归图像生成模型。我们将常用的 k-means 方法与我们提出的 DCPE 方法进行比较，如 Table I 所示。通过应用聚类，我们将码本大小减少了一半，并采用 Section IV-A 中描述的推理策略。这种词汇缩减减少了自回归模型输入和输出层的参数数量。从数值结果来看，显然使用 k-means 聚类训练的模型与基线相比表现下降。这种性能下降可能是由于 k-means 无法捕捉标记相似性信息，从而阻碍了模型对训练数据的拟合能力。相比之下，我们的 DCPE 方法通过利用判别性码本先验获得了更好的结果。值得注意的是，在 LlamaGen-B 的实验中，尽管使用了更少的参数和更小的词汇量，我们的方法仍然优于基线。我们将此归因于小型自回归模型在拟合复杂标记分布时面临的困难，而由 DCPE 提取的码本先验使得收敛更容易。在更大的模型上，我们的方法也维持了与基线相当的性能。

为了证明上述收敛加速效果，我们比较了不同时期普通和基于聚类的 LlamaGen-B 的性能，如 Fig. 3 所示。结

TABLE I
香草和基于聚类的 LLAMA GEN [?] 在 256×256 分辨率下的类条件 IMAGENET 上的性能。

Method	# Para.	# Vocab.	FID↓	IS↑	Precision↑	Recall↑
LlamaGen-B [†] [?]	111M	16384	5.29	185.7	0.84	0.45
+ k-means	94M	8192	5.56	188.4	0.83	0.44
+ DCPE	94M	8192	4.83	198.8	0.82	0.47
LlamaGen-L [†] [?]	343M	16384	3.68	248.9	0.83	0.52
+ k-means	311M	8192	4.08	216.7	0.79	0.55
+ DCPE	311M	8192	3.34	238.4	0.81	0.54
LlamaGen-XL [†] [?]	775M	16384	3.14	269.9	0.83	0.54
+ k-means	719M	8192	3.32	240.1	0.79	0.58
+ DCPE	719M	8192	2.86	267.4	0.82	0.56

[†] indicates results reproduced using the official code. “# Vocab.” denotes the codebook size for the baseline and the number of clusters for cluster-based methods. Our DCPE achieves comparable or even better performance than the baseline using fewer parameters and a smaller vocabulary size, especially on smaller models, while k-means shows degradation due to its inferior codebook prior.

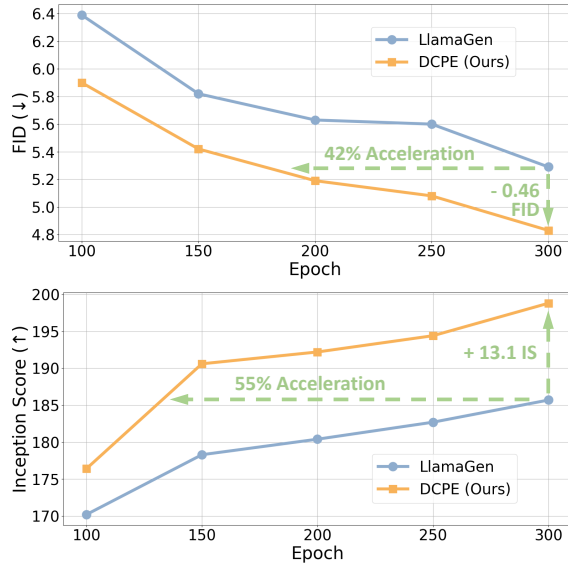


Fig. 3. 原生 LlamaGen-B 在不同时期的性能表现及其基于我们 DCPE 的聚类训练。DCPE 提取的码本先验可以有效加速模型的收敛。

果表明，通过 DCPE 获取的聚类进行训练显著加速了自回归模型的收敛。例如，为了达到与基线相同的 FID 和 IS，使用 DCPE 进行训练分别实现了 42 % 和 55 % 的加速。当训练相同数量的时期时，我们的方法使 FID 降低了 0.46，并使 IS 提高了 13.1。

尽管 CTF [?] 由于次优的 k-means 使用需要训练一个大型的精细化模型 (311M 参数)，我们的 DCPE 即便没有精细化模型也表现良好，如 Table I 中所示。我们进一步进行实验，训练一个类似 CTF 的精细化模型，将预测的簇映射到真实的标记。但由于我们方法的强大基线表现，我们只训练一个小的单层注意力网络，不像 CTF 那样。如 Table II 中所示，利用一个拥有 25M 参数的小型精细化模型，我们实现了进一步的性能提升，特别体现在提升的 IS 分数上，这表明图像视觉质量更好。然而，我们观察到 FID 有轻微下降，我们将其归因于精细化模型导致的分布偏移，因为 FID 衡量生成的和训练图像分布之间的差异。值得注意的是，在 LlamaGen-L 和 LlamaGen-XL 上，使

TABLE II
小型精炼模型在 DCPE 上的表现。

Method	# Para.	FID↓	IS↑	Pr.↑	Re.↑
DCPE	94M	4.83	198.8	0.82	0.47
+ refine	94+25M	5.02	204.5	0.84	0.45
DCPE	311M	3.34	238.4	0.81	0.54
+ refine	311+25M	3.41	248.7	0.83	0.52
DCPE	719M	2.86	267.4	0.82	0.56
+ refine	719+25M	3.11	273.4	0.83	0.54

Evaluations are conducted under the same settings as in Table I. “+ refine” means a small one-layer attention network is trained to refine the predicted clusters into tokens, similar to what CTF [?] does. Explanations of the performance gap can be found in Section IV-B.

用精细化模型的结果开始超越原始训练的性能 (Table I)，尽管使用更少的参数和更小的词汇表。我们还训练了具有较少簇的精细化模型 (见 Section IV-C)，发现这样的单层网络在大的码书下采样率下表现良好。

为了进一步证明 DCPE 可以在具有不同 token 分布的图像分词器中提供改进，我们将原有的图像分词器替换为更大的 GigaTok [?]，如 Table III 所示。由于分词器的规模显著扩大 ($10 \times$ 参数)，缩减的码本已无法通过补充细节来提高性能。然而，结果仍然显示 DCPE 在不同词汇量下优于 k-means。值得注意的是，随着词汇量的减少，性能差距变得更加显著。这表明，随着簇的增大，k-means 更可能生成包含不同语义 token 的簇。相比之下，DCPE 的凝聚设计更好地保留了每个簇中的语义一致性。

我们将我们的 DCPE 集成到 IAR 流水线中，以替代默认的 k-means 聚类。实现细节在 Section III-C 和 IV-A 中描述。如 Table IV 所示，用我们的 DCPE 替换 k-means 聚类有效地提高了 IAR 的性能。这种改进可以归因于 DCPE 能够提取更具判别能力的码本先验，从而为自回归生成模型的训练提供更好的辅助信息。

C. 消融研究与分析

正如在 Section III-A 中讨论的，我们将 k-means 在从令牌特征空间中提取信息的次优表现归因于两个关键问题：令牌空间差异和质心距离不准确。为了解决这些问

TABLE III
使用图像标记器 GigaTok-B-L 的 DCPE 性能 [?]。

Method	# Vocab.	FID↓	IS↑	Pr.↑	Re.↑
GigaTok [†] [?]	16384	3.39	263.7	0.81	0.55
+ k-means	8192	3.67	259.8	0.79	0.56
+ DCPE	8192	3.65	261.5	0.80	0.55
+ k-means	4096	4.74	240.2	0.77	0.55
+ DCPE	4096	4.19	249.3	0.78	0.56

Evaluations are conducted under the same settings as in Table I. [†] indicates results reproduced using the official code. Our DCPE consistently outperforms k-means clustering across different vocabulary sizes.

TABLE IV
将 DCPE 集成到 IAR 在类条件 IMAGENET 上的性能比较, 分辨率为 256 × 256。

Type	Model	# Para.	FID↓	IS↑	Pr.↑	Re.↑
GAN	BigGAN [?]	112M	6.95	224.5	0.89	0.38
	GigaGAN [?]	569M	3.45	225.5	0.84	0.61
	StyleGAN-XL [?]	166M	2.30	265.1	0.78	0.53
Diff.	ADM [?]	554M	10.94	101.0	0.69	0.63
	CDM [?]	-	4.88	158.7	-	-
	LDM-4 [?]	400M	3.60	247.7	-	-
	DiT-XL/2 [?]	675M	2.27	278.2	0.83	0.57
Mask.	MaskGIT [?]	227M	6.18	182.1	0.80	0.51
	MaskGIT-re [?]	227M	4.02	355.6	-	-
VAR	VAR-d16 [?]	310M	3.30	274.4	0.84	0.51
	VAR-d20 [?]	600M	2.57	302.6	0.83	0.56
	VAR-d24 [?]	1.0B	2.09	312.9	0.82	0.59
AR	VQGAN [?]	227M	18.65	80.4	0.78	0.26
	VQGAN [?]	1.4B	15.78	74.3	-	-
	VQGAN-re [?]	1.4B	5.20	280.3	-	-
	ViT-VQGAN [?]	1.7B	4.17	175.1	-	-
	ViT-VQGAN-re [?]	1.7B	3.48	175.1	-	-
	RQTran. [?]	3.8B	7.55	134.0	-	-
	RQTran.-re [?]	3.8B	3.80	323.7	-	-
	LlamaGen-B [†] [?]	111M	5.29	185.7	0.84	0.45
	LlamaGen-L [†] [?]	343M	3.68	248.9	0.83	0.52
IAR	LlamaGen-XL [†] [?]	775M	3.14	269.9	0.83	0.54
	IAR-B [?]	111M	5.14	202.0	0.85	0.45
	+ DCPE	111M	5.12	209.5	0.86	0.44
	IAR-L [?]	343M	3.18	234.8	0.82	0.53
	+ DCPE	343M	3.14	249.5	0.83	0.53
	IAR-XL [?]	775M	2.52	248.1	0.82	0.58
	+ DCPE	775M	2.49	270.8	0.83	0.58

[†] indicates results reproduced using the official code. Replacing the default k-means clustering with our DCPE improves the performance of IAR.

题, 我们分别通过采用凝聚聚类和无质心距离计算来进行实验, 如 Table V 所示。除了基线 k-means 及其常见变体 k-means++, 我们还独立评估了凝聚聚类和无质心距离计算的效果。为了隔离凝聚聚类的影响, 我们在我们的 DCPE 中去除了基于实例的距离的使用。相反, 为了单独测试无质心距离计算, 我们将其整合到朴素 k-means 算法中, 其他组件保持不变。结果表明, 这两种技术都有助于更有效地提取代码本的先验, 从而提高自回归模型的性能。虽然基于密度的聚类方法如 DBSCAN [?] 也可能缓解上述问题, 但由于它们无法控制聚类的数量, 使得它们在我们的任务中不切实际, 因此我们没有将它们纳入实验。

1) 簇中的符号相似度: 我们的 DCPE 在增强集群内标记语义相似度方面的有效性, 如 Section III-B 中所讨论, 通过重建实验进一步得到验证。在这里, 我们通过 LlamaGen tokenizer [?] 获得的所有图像标记用同一集

TABLE V
对两种提议设计的消融研究。

Method	C-F	Agglo.	FID↓	IS↑
k-means	×	×	5.56	188.4
k-means++	×	×	5.33	187.9
DCPE w/ centroid	×	✓	5.25	197.5
k-means w/o centroid	✓	×	5.01	189.3
DCPE	✓	✓	4.83	198.8

Evaluations are conducted under the same settings as in Table I. “C-F” means centroid-free and “Agglo.” means agglomerative clustering. Both two main components of our DCPE contribute to improved generation performance.

群的标记进行替换, 然后将它们传递到解码器中重建图像。具体来说, 我们从 ImageNet 验证集中选择了 50,000 张图像, 并对它们进行中心裁剪。重建后的图像与原始图像进行比较, 计算在不同的聚类方法下的 rFID、PSNR 和 SSIM [?], [?]。由于官方 LlamaGen 代码库没有提供 PSNR 和 SSIM 评估的实现, 我们遵循常用做法, 并利用 Python 包 scikit-image 中的相应功能进行评估。使用开源 LlamaGen 图像标记器以及我们的评估实施, 我们能够重现 LlamaGen 论文中报告的 rFID 和 PSNR 值。然而, SSIM 结果与原报告相比存在差异。为确保公平比较, 我们报告了我们复现的结果。

Table VI 中的结果表明, 与 k-means 相比, 使用由我们的 DCPE 产生的聚类导致图像质量的下降更少。随着词汇量的减少, 即聚类规模的增大, 这一优势变得更加明显。结果表明, 我们的 DCPE 更好地在聚类内保持语义一致性, 允许被扰动的图像标记被正确解码。

2) 簇大小分析: 为了证明我们的 DCPE 能够有效处理不均匀分布的标记特征空间, 我们分析了由 k-均值聚类和我们的 DCPE 生成的聚类大小分布, 如 Fig. 4 所示。为了更好地进行可视化, 我们将用于 LlamaGen 的 16k 大小的码书聚类成 128 个聚类。结果表明, k-均值形成的聚类表现出相对均匀的大小, 这与 Section III-A 中讨论的标记特征空间的固有稀疏性相矛盾。相比之下, DCPE 产生了大小范围更广的聚类。这种变异归因于我们的聚合聚类策略, 该策略优先合并相似的标记以确保组内相似性, 并避免 k-均值聚类中的同步更新, 这常常忽略密度的变化。

3) 不同词汇量下的表现: 我们在不同的词汇大小下评估我们的 DCPE, 即聚类的数量, 并为每种设定训练一个精细模型。结果如 Table VII 所示。由于较小的词汇大小需要更少的输入和输出层参数, 因此, 在 “# Para.” 列中报告的模型参数数量相应变化。结果显示, 使用较小词汇大小训练的模型往往表现较差。这种退化发生是因为同一聚类中的标记随着聚类大小的增加在语义上变得更加多样化, 导致生成图像中的细节损失更大。然而, 我们发现, 引入一个由单层注意机制组成的轻量级精细模型可以显著改善性能。例如, 当词汇大小为 2048 时, 精细模型取得了与默认设置相当的性能, 且使用的参数更少。然而, 当词汇大小过小时, 比如 1024, 聚类内过多的语义混合增加了生成模型的收敛难度。

在 Section IV-B 中, 我们讨论了与 CTF 不同的是, 我们仅训练了一个一层的注意力网络, 这得益于我们的 DCPE 提供的强大基线性能。我们进一步进行了实验, 以探讨使用更大的精细化模型是否能提高基于聚类的生成模型的性能。如 Table VIII 所示, 增大精细化模型的规模仅带来微

TABLE VI
替换标记后的图像重建性能。

Method	# Vocab.	rFID↓	PSNR↑	SSIM↑
w/o replacement [†]	16384	2.20	20.69	0.514
w/ k-means	8192	4.14	19.57	0.471
w/ DCPE	8192	3.48	19.78	0.479
w/ k-means	4096	8.27	18.90	0.442
w/ DCPE	4096	7.23	19.01	0.447

We replace image tokens extracted by the LlamaGen tokenizer with random tokens from the same cluster (see Section IV-C). [†] indicates results reproduced with the official code. Using clusters from our DCPE shows less performance degradation due to consistent token semantics in each cluster.

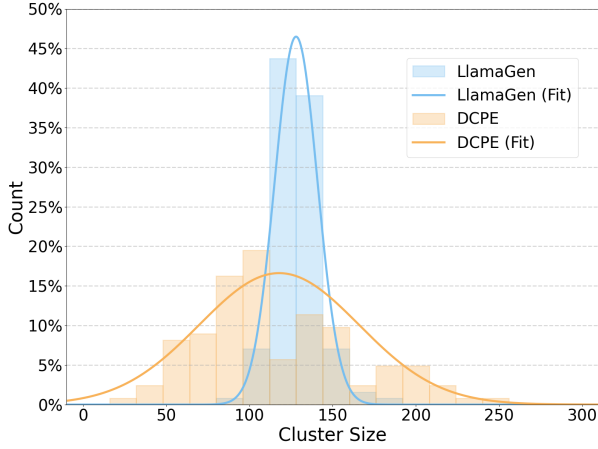


Fig. 4. 通过 k-means 聚类我们的 DCPE 获得的聚类大小。我们从大小为 16k 的码本中导出 128 个聚类，并使用高斯分布来拟合它们以获得更好的可视化效果。我们的 DCPE 产生了具有更大范围大小的聚类，这与 token 密度的固有稀疏性一致。

小的性能提升。这是因为，在我们的设置（词汇量 = 8192）下，每个聚类内的标记表现出高度的相似性，随机选择已经足够。因此，将聚类索引细化为标记索引所带来的好处有限。

D. 超参数选择

为了验证随机选择策略在对训练于聚类索引的自回归模型进行适当推理时的有效性，我们进行了使用不同随机选择种子的实验。然后，我们评估了生成图像的质量，以评估这一生成范式的稳定性。如 Table VIII 所示，生成图像的评价指标在不同的随机种子下保持一致，仅观察到小的标准差。这些结果表明，我们的 DCPE 生成的每个聚类内的标记具有高度的语义相似性。因此，随机选择的标记传达了类似的语义内容，导致在最终解码图像中表现的变化很小。

与之前的研究类似 [?], [?], [?], [?]，我们在不同的 Classifier-Free Guidance (CFG) 设置下报告生成指标，如 Fig. 5 所示。与现有的发现一致，较低的 CFG 值产生更好的 FID 分数，表明与训练数据分布更紧密的对齐。相反，较高的 CFG 值导致更高的 inception 分数，反映了图像质量的提升。为了公平比较，我们遵循之前的工作 [?], [?], [?]，将 LlamaGen-B 的默认 CFG 值设置为 2.0，这提供了 FID 和 IS 之间的良好权衡。

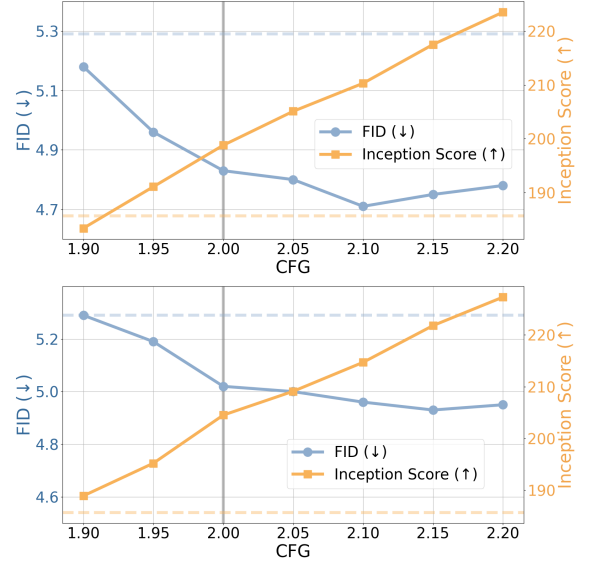


Fig. 5. 在不同的无分类器引导 (CFG) 设置下，DCPE (上) 和 DCPE+Refine (下) 在 LlamaGen-B 上的表现。虚线表示普通 LlamaGen-B 的结果。我们选择 CFG = 2.0 (以灰线突出显示) 作为默认值，因为其表现相对均衡。

正如在 Section III-C 中讨论的，我们将 DCPE 整合到 IAR 的训练框架中，并报告了更优越的结果。我们进一步对 IAR 中的两个关键超参数进行了消融研究：簇的数量和簇导向交叉熵损失的损失权重。结果如 Table X 中所示，表明我们的默认设置，即 512 个簇和权重为 1.0 的损失，在不同配置下都能产生相对较强的性能。

E. 可视化结果

我们在 Fig. 6 中展示了使用 LlamaGen-XL 与我们的 DCPE 和 CFG 为 4.0 生成的图像，遵循常见的设置 [?], [?], [?]。图像生成类别包括：秃鹰、小熊猫、船屋、城堡、双桅船、珊瑚礁、冰淇淋和火山。附录中提供了其他生成样本。

V. 局限性和未来工作

由于资源限制，我们仅使用 LlamaGen 作为基线进行了类条件图像生成实验。最近，出现了越来越多的自回归图像生成范式，例如使用下一个尺度预测的 VAR [?]，使用随机下一个标记预测的 RAR [?]，以及使用下一个邻居预测的 NAR [?]。在这些新兴范式上评估我们 DCPE 方法的有效性同样具有价值。此外，文本条件的图像生成具有更大的实际意义，值得进行更广泛的探索。

如 Section III-C 所讨论，通过 DCPE 获得的簇在大小上有所不同。为了确保在将其整合到 IAR [?] 中的正确训练，我们将簇的 logits 重新定义为簇中所有 tokens 的 logits 的平均值，而不是简单的求和。此修改在反向传播过程中引入了不同簇中 tokens 梯度的不一致，这可能导致次优的结果。我们没有进一步研究解决这个问题的方法，但一个有希望的研究方向是探索增强训练稳定性的方法 [?], [?], [?], [?], [?]

关于算法设计，DCPE 采用贪心策略，通过在每一步合并两个最近的簇。这种方法仅优化当前迭代，并不保证获得全局最优解。它可能在特定的标记分布下失败。考虑到

TABLE VII
在 LLAMAGEN-B 上, DCPE 在不同词汇量下的性能

Method	# Para.	# Vocab.	FID↓	IS↑	Precision↑	Recall↑
LlamaGen-B [†] [?]	111M	16384	5.29	185.7	0.84	0.45
DCPE	94M	8192	4.83	198.8	0.82	0.47
+ refine	94M+25M	8192	5.02	204.5	0.84	0.45
DCPE	92M	4096	5.52	191.8	0.80	0.48
+ refine	92M+24M	4096	5.66	207.0	0.84	0.43
DCPE	89M	2048	9.44	161.5	0.73	0.50
+ refine	89M+22M	2048	5.85	208.8	0.82	0.45
DCPE	87M	1024	18.70	111.0	0.63	0.49
+ refine	87M+21M	1024	7.21	193.5	0.80	0.41

“+ refine” means a small one-layer attention network is trained to refine the predicted clusters into tokens. Smaller vocabulary sizes tend to perform worse, while a small refine model can effectively improve performance, making a model with less parameter and smaller vocabulary perform comparably to the baseline.

TABLE VIII
DCPE 在 LLAMAGEN-B 上使用不同精炼模型大小的性能

Method	# Layers	# Para.	# Vocab.	FID↓	IS↑	Precision↑	Recall↑
LlamaGen-B [†] [?]	-	111M	16384	5.29	185.7	0.84	0.45
DCPE	-	94M	8192	4.83	198.8	0.82	0.47
+ refine	1	94M+25M	8192	5.02	204.5	0.84	0.45
+ refine	3	94M+41M	8192	5.18	204.7	0.84	0.45
+ refine	6	94M+62M	8192	5.21	204.9	0.85	0.45
+ refine	9	94M+83M	8192	5.20	205.8	0.85	0.45
+ refine	12	94M+105M	8192	5.19	206.3	0.85	0.45

“# Layers” indicates the number of attention layers in the refine model. Default settings are highlighted in gray. Scaling up the refine model leads to only marginal performance gains, because our DCPE generate high intra-similarity clusters, reducing the need for refinement.

TABLE IX
使用不同的随机选择种子进行生成。

Method	# Vocab.	Seed	FID↓	IS↑
LlamaGen-B [†] [?]	16384	0	5.29	185.7
DCPE	8192	0	4.83	198.8
		1	4.92	199.1
		2	4.93	195.4
		3	4.84	197.6
		4	4.84	198.7
Std.	-	-	0.044	1.359

Evaluations are conducted under the same settings as in Table I. The default settings are highlighted in gray. The results show minimal variation due to the high intra-similarity clusters generated by our DCPE.

找到一个全局最优聚类策略是 NP 难题, 未来的研究可以探索更有效的方法来逼近全局最优。

VI. 结论

在这项工作中, 我们研究并揭示了现有方法中利用简单的 k 均值聚类在代码簿标记上促进自回归图像生成时出现的标记空间差异和质心距离不准确问题。为了提供更好的代码簿先验, 我们提出了判别代码簿先验提取器 (DCPE)。它用更可靠的实例距离替代不可靠的质心距离, 以反映真实的标记相似性。通过一种聚合合并策略, DCPE 进一

TABLE X
关于在 IAR 中与 DCPE 一起使用的超参数的消融研究。

# Cluster	λ	FID↓	IS↑	Precision↑	Recall↑
512	1.0	5.12	209.5	0.86	0.44
512	0.2	5.15	202.6	0.85	0.45
	0.5	5.37	200.7	0.85	0.43
	2.0	5.33	198.2	0.86	0.43
128	1.0	5.29	204.9	0.86	0.43
256		5.15	203.6	0.85	0.44
1024		5.23	202.3	0.85	0.45

Evaluations are conducted under the same settings as LlamaGen-B in Table IV. “# Cluster” refers to the number of clusters, and λ indicates the weight the cluster-oriented cross-entropy loss. Results obtained under the default setting are highlighted in gray.

步通过保留高密度区域来解决标记空间差异问题。大量实验表明, DCPE 是即插即用的, 并能无缝集成到现有代码簿先验基础的范式中。通过有效利用嵌入的标记相似性, DCPE 改善了生成质量并加速了训练。



Fig. 6. 使用我们的 DCPE 和 CFG 为 4.0，通过 LlamaGen-XL 生成的图像。附录中提供了更多生成的样本。