Token Representation Shrinkage Impairs Creativity of Generative Models

Anonymous Author(s)
Affiliation
Address
email

Abstract

Transformer-based generative models have been widely used for generating high-quality images and other continuous data modalities. Despite their widespread adoption, these models frequently exhibit limitations in creativity, often failing to produce diverse and novel outputs. Most existing studies analysing these shortcomings have predominantly concentrated on enhancing the generative architecture or training methodologies. In contrast, our study shifts the focus to the tokenization process, exploring how discretizing continuous representations into discrete tokens influences the overall creativity of generative models. Through systematic analysis, we identify a critical phenomenon we term "token representation shrinkage," characterized by the collapse of representation diversity within discrete codebook tokens and their continuous latent embeddings in vector quantization, which is one of the most popular discrete tokenization method used. Our findings reveal that this shrinkage problem significantly reduces the creativity of generative models, adversely affecting performance across various domains, including natural images and real-world medical images.

1 Introduction

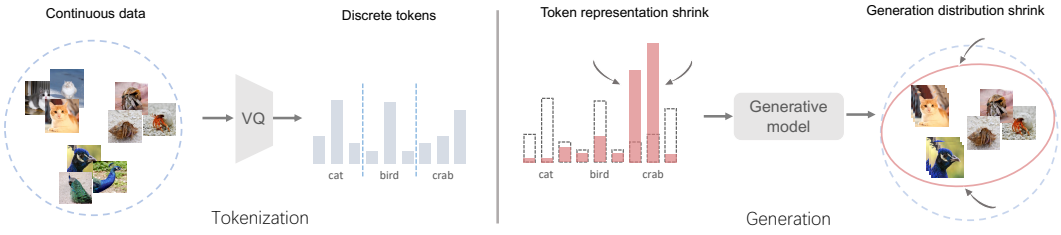


Figure 1: **Token representation shrinkage leads to diversity loss in generative model.** Left: Vector quantization is a widely used technique to map continuous data into discrete token which enable the generative model's generation. Right: We observe that token representation shrinkage, manifested as narrow distribution in latent space, leads to a shrunk distribution of the generated data.

Transformer-based generative models for autoregressive generation have gained significant popularity in recent years in the field of image generation. These models underpin many state-of-the-art systems such as DALL-E [3] and VAR [25], which have found wide-ranging applications in art creation, design automation, and data augmentation. Their practical value lies not only in producing visually compelling images but also in enabling new workflows for creative and industrial domains.

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

Despite their success, transformer-based generative models suffer from a widely observed issue: the synthetic images they generate often exhibit a narrower distribution compared to original images. This phenomenon, commonly referred to as mode collapse, results in limited diversity in the generated content. Mode collapse leads to a loss of diversity in generated outputs, causing the model to ignore valid variations in the data distribution, which limits its generalization, realism, and utility in downstream tasks. In this study, we refer the ability of generative models to produce diverse high quality outputs as creativity. Therefore, mode collapse and limited diversity in output will lead to decreased creativity of the generative models.

Most existing studies on these problems have predominantly focused on the generative architecture or training methodologies. To address this, various studies have proposed architectural innovations or alternative training objectives. For example, VQGAN [10] incorporates vector quantization to learn a diverse discrete codebook, while ImageGPT [5] treats images as sequences of pixels to better capture complex data distributions and enhance generative diversity.

However, in this work, we identify a previously overlooked but critical factor in tokenization, termed token representation shrinkage, which contributes to the decline in generative creativity. Specifically, the root of this problem lies in a core component of transformer-based image generators: the use of vector quantization (VQ), one of the most widely used discrete tokenizers, for tokenizing images. VQ is crucial for converting continuous image features into discrete tokens suitable for transformer processing. However, we find that when the token representation distribution undergoes shrinkage, the generative model's output creativity is significantly reduced. As shown in Fig. 1, VQ techniques map continuous data into discrete tokens. However, when tokens shrank into a limited region of the distribution, the generated outputs are also constrained to a narrow portion of the data space, resulting in reduced diversity and diminished modality coverage.

We further identify a specific mechanism that contributes to token representation shrinkage: the commonly used token initialization strategy during VQ training. Typically, token embeddings are initialized based on the outputs of an untrained encoder, which results in a clustered initial token distribution. This initialization bias suppresses the token space's ability to expand during training, preventing it from aligning with the true data distribution and thus inducing representation shrinkage.

To address this, we propose a simple yet effective solution: pretrain the encoder without VQ and then fine-tune it with VQ enabled. This approach allows the encoder to learn meaningful semantic representations before quantization is introduced, thereby reducing the resistance faced during VQ optimization and alleviating the token shrinkage effect. We validate our hypothesis and proposed method through extensive experiments on both synthetic datasets and real-world datasets, including ImageNet, CIFAR-10, and the Ocular Disease Recognition medical dataset. Our results demonstrate that token representation shrinkage leads to decreased generative creativity and that our approach significantly mitigates this issue, improving both diversity and fidelity of generated images.

Our main contributions are summarized as follows:

- We identify a previously underexplored cause of mode collapse in transformer-based generative models: token representation shrinkage.
- We provide a detailed analysis of how poor token initialization contributes to this phenomenon.
- We propose a simple and effective training strategy, pretraining without VQ followed by fine-tuning with VQ, to resolve the issue.
- We empirically validate our findings on both synthetic and real-world datasets, demonstrating improved generative performance.

2 Related Works

Vector Quantization is foundational in data compression and signal processing per Shannon's rate-distortion theory [12, 7] , traditionally relied on methods like K-means clustering [19] but faced high complexity with high-dimensional data [17]. To mitigate this challenge, DeepVQ [17] improved efficiency by mapping data to lower-dimensional latent spaces before quantization. Moreover, [26] proposed VQ-VAE which integrates VQ with variational autoencoders, using a straight-through estimator [2] to handle discrete variables. To refine VQ methods for improved performance, variants

such as Residual Quantization [18], Product Quantization [6], and Soft Convex Quantization [11] further enhanced representation capacity and efficiency. Recent advances incorporate attention mechanisms and transformer architectures [27, 28] to dynamically select codebooks and capture global data dependencies. Recent works also explore per-channel codebooks [14] and neural network variants of residual quantization [15] to predict specialized codebooks, enhancing the model's expressive power.

VQ has been widely applied across various domains. In natural language processing, VQ facilitates sequence modeling [16] enhancing tasks such as language modeling. In computer vision, VQ has significantly advanced image generation and compression techniques [10]. Similarly, in audio processing, VQ techniques have captured complex temporal dependencies [8]. Furthermore, in multimodal applications, VQ supports the integration of different data types through shared discrete representations [23].

Despite these advancements, VQ methods encounter challenges that restrict their broader application, including but not limited to codebook collapse, training instability, and computational overhead. Extensive research has been conducted on solving the codebook collapse problem, where only a subset of tokens are used leading to inefficient representation usage and reduced diversity in outputs, by reducing token dimension [28], orthogonal regularization loss [24], multi-headed VQ [20], finite scalar quantization [22], and Lookup Free Quantization [29]. Recent methods like [13] and [1] also strive to enhance tokens usage efficiency. However, beyond the widely recognized issue of codebook collapse, our work identifies, investigates, and proposes potential solutions for collapses of tokens and reconstruction, which pose serious challenges to VQ and merit attention.

3 Preliminary

3.1 Definition of Creativity for Generative Model

In this study, we define the *creativity* of a generative model as the diversity of high-quality content it generates. For example, an ideal image generative model should produce high-fidelity images which are very different from each other. Most previous works related to creativity of generative models focus their research on the generative models [10, 5]. However, we observe that shrinkage of token representation distribution is also an important factor to consider for creativity. Our experiments suggest that **token representation shrinkage** significantly impairs the creativity of transformer-based generative models.

3.2 Preliminary of Vector Quatization

VQ-VAE We define the VQ-VAE as following: an encoder E_θ, a decoder D_θ and a set of tokens $\mathcal{T} = \{t_1, t_2, \ldots, t_S\}$. The token set $\mathcal{T}$ constitutes the codebook, which is employed to store the discretized representations. The encoder is responsible for mapping the raw data $X = \{x_1, x_2, \ldots, x_N\}$ to a set of continuous representations $\mathcal{Z} = E_\theta(X)$, where $\mathcal{Z} = \{z_1, z_2, \ldots, z_N\}$. And the decoder reconstructs the data $X' = D_\theta(\hat{Z})$ based on the set of discretized representations $\hat{Z}$, where $\hat{Z} = \{\hat{z}_1, \hat{z}_2, \ldots, \hat{z}_N\}$.The process of tokenizing a continuous representation z_j to discrete representation $\hat{z_j}$ is as following:

$$\hat{z_j} = \arg\min_{t_k \in \mathcal{T}} \|z_j - t_k\|, \tag{1}$$

where t_k is a token in token set $\mathcal{T}$ and k is the index. This quantization is performed by finding the nearest token t_k in $\mathcal{T}$.The optimization objective comprises reconstruction loss $\mathcal{L}_{\text{recon}}$, codebook loss $\mathcal{L}_{\text{codebook}}$, and commitment loss $\mathcal{L}_{\text{commit}}$. Additionally, we adopt the exponential moving averages (EMA) adopted by [26] to update the codebook instead of the codebook loss term.

Initialization Strategy For codebook initialization, a widely used initialization strategy is K-means[30]. It uses the encoder output $\mathcal{Z}$ and perform K-means algorithm to initialize the tokens $\mathcal{T}$, where N is the number of encoder output and S is the number of tokens. The initialization aims to minimize the total distance from each vector z_j to its nearest token t_k. The optimizing function is shown in equation 2,

$$\min \sum_{j=1}^{N} \sum_{k=1}^{S} r_{jk} \|z_j - t_k\|^2, \tag{2}$$

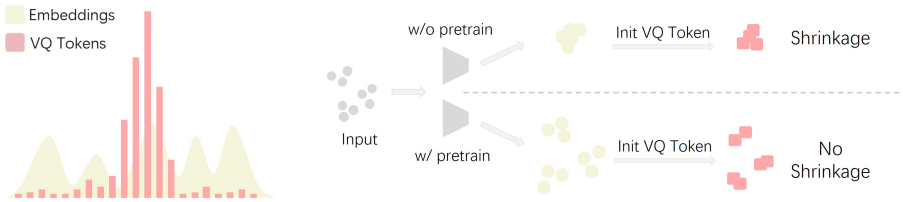


Figure 2: **Token representation shrinkage phenomena is attributed to biased initialization.** (a) Token representation shrinkage refers to the phenomenon where token becomes concentrated on a small number of modes, despite the original continuous embeddings exhibiting diverse and well-separated modes. (b) Our analysis suggests that token representation shrinkage arises from initializing tokens with untrained embeddings that lack sufficient modality information.

where $r_{jk} = 1$ if z_j is assigned to cluster center t_k, otherwise $r_{jk} = 0$.

4 Token Representation Shrinkage Problems

This section presents an analysis of the token representation shrinkage phenomenon and investigates its underlying causes.

4.1 Shrinkage Phenomena and Sythentic Experiments Results

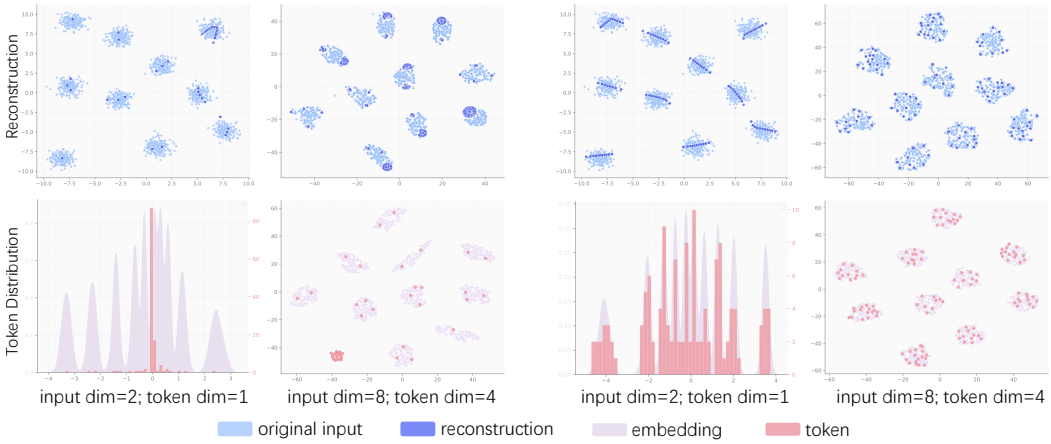
Token representation shrinkage is characterized by a disproportionate concentration of tokens around a limited subset of encoder output embeddings, as shown in Fig. 2 (a). This shrinkage results in a poor representation since the ideal scenario requires a fitting distribution of tokens that effectively aligns with the underlying embedding space.

To validate the token representation shrinkage phenomenon, we conduct experiments on our synthetic dataset using VQ-VAE. Specifically, we use VQ-VAE to reconstruct the input data and compare the resulting token distribution with the original data distribution. The synthetic dataset comprises 10,000 data points, uniformly sampled from 10 distinct Gaussian distributions (see Sec. 5.1 for details). As shown in Fig. 3 (a), (c), and (e), tokens densely cluster within a specific region of the latent space, which subsequently causes the reconstructed data to collapse. As a result, the reconstructions fail to capture the full modality spectrum of the original data.

One contributing factor to token representation shrinkage is the clustering of token embeddings during codebook initialization. This occurs when the initial embeddings are distributed within a narrow region of the latent space, limiting their expressiveness and leading to early-stage shrinkage. As shown in Fig.2 (b), the output distribution of an untrained encoder is significantly more concentrated compared to that of a trained encoder.

In order to examine how untrained encoder initialization contributes to token representation shrinkage, we compare the embedding distributions produced by trained and untrained encoders on the synthetic dataset. We observe that the untrained encoder produces embeddings that are concentrated in a narrower region and exhibit fewer distinct peaks, suggesting that they represent fewer, less distinguishable modes. This supports the conclusion that token representation shrinkage is primarily caused by the use of untrained encoders for token initialization. Since the untrained encoder lacks the capacity to extract meaningful features from the input data, it maps diverse inputs to similar embeddings, leading to a poorly distributed token initialization and reduced representational diversity. Further experimental details and visualizations are provided in the supplementary material.

Building on these observations, we hypothesize that if tokens are initialized based on encoder that has learned semantic distinctions and its output embeddings are dispersed, it would enhance the semantic distinction among tokens and thus control token representation shrinkage. Consequently, we propose a straightforward yet effective method to mitigate token representation shrinkage: pretrain without VQ, then fine-tune with VQ. It first trains an autoencoder, and then trains the VQ-VAE initialized with the weights of the autoencoder trained at the first stage. Pretraining the encoder allows it to



(a) w/ shrinkage (b) w/o shrinkage

Figure 3: **Visualization of token shrinkage effects in synthetic experiments**. (a) With shrinkage: most of the token distribution clusters into a narrow region of the embedding space, leading to a loss of diversity across reconstruction modes under different input dimensions. (b) Without shrinkage: tokens are well distributed across the embedding space, enabling accurate and diverse reconstructions.

discern differences in input data, resulting in more distinctly spaced embeddings, providing a robust foundation for initializing the tokens, as demonstrated in Fig. 2 (b).

We evaluate the effect of our pretraining strategy on the synthetic dataset, with results shown in Fig. 3. The comparison between subfigures (a) and (b) demonstrates that when the shrinkage problem is mitigated by our pretraining approach (Fig. 3 (b)), the resulting token distribution becomes more uniform, and the reconstruction aligns more closely with the original input distribution. Notably, under higher input dimensions (input dim=8), the token shrinkage problem still leads to degraded reconstruction performance. In contrast, the version without shrinkage produces a reconstruction distribution that aligns more closely with the original data distribution. This suggests that addressing token shrinkage is critical for enhancing the creativity (diversity and quality) of generative models.

4.2 Formal Definition of Token Representation Shrinkage

To mathematically analyze the token representation shrinkage effect, we consider a data distribution constructed from K well-separated and equally weighted component distributions $p(x|k)$,

$$p(x) = \sum_{k=1}^{K} p(x|k)p(k) = \frac{1}{K}\sum_{k=1}^{K} p(x|k), \tag{3}$$

where $p(k) = \frac{1}{K}$ because of equal weights. For simplicity, we assume that both the encoder and decoder are identity mapping (i.e. $X' = Dec(Enc(X)) = X$), and that the transformer can perfectly model the full token distribution. Under these assumptions, the only source of distortion arises from *vector quantization*. Accordingly, the expected mean squared error in pixel space roughly express the upper bound of generation quality:

$$\mathcal{E} = \mathbb{E}_{x\sim p}\left[\|q(x) - x\|_2^2\right], \tag{4}$$

where q is quantization function. We assume the entropy of the generated mode distribution measure the diversity:

$$H = -\sum_{k=1}^{K} p_k \log p_k, \quad \text{where} \quad p_k = \frac{|T_k|}{\sum_{j=1}^{K}|T_j|}, \quad T_k = \{t_i \mid t_i \in \text{cluster } k\}, \tag{5}$$

where T_k is the set of tokens assigned to cluster k, and p_k is the empirical probability (proportion) of tokens in that cluster.

In the ideal case of balanced token utilization, we expect $|T_k| \approx S/K$, yielding $p_k \approx 1/K$, maximal entropy $H = \log K$, and minimal quantization error Q. However, under **token representation shrinkage**, token becomes concentrated in a subset of modes $\mathcal{J} \subset \{1, \ldots, K\}$, where $|\mathcal{J}| = M \ll K$. This leads to a reduced entropy

$$\Delta\mathcal{E} = logM - logK < 0, \tag{6}$$

which means the diversity will be impaired. Moreover, samples from inactive modes $k \notin \mathcal{J}$ are forced to encode using distant tokens, thereby increasing the quantization error and subsequently decreasing the generation quality.

5 Experiments Design and Results

In this section, we firstly conduct experiments on CIFAR-10 to validate the existence of token representation shrinkage in the real-world dataset. And then we demonstrate that token representation shrinkage negatively impacts the creativity of generative models, thereby decreasing both the diversity and fidelity of generated samples.we conduct experiments on two representative generative models, MaskGIT [4] and VAR [25], using both the ImageNet-100 dataset and a medical image dataset.

It is important to note that in the experiments involving generative models, the use of GAN-based losses can introduce smoothing effects to the model, potentially hallucinating the presence of token representation shrinkage. Therefore, in this section, we adopt VQ-VAE as the image tokenizer. The training loss includes *codebook loss*, *commitment loss*, *MSE loss*, and *perceptual loss*. Generative experimental results based on VQGAN are available in the supplementary.

5.1 Experiment Setup

Dataset As mention in in Sec. 4.1, we conduct experiments on a synthetic dataset to validate our hypothesis regarding the causes of token representation shrinkage. The synthetic dataset consists of 10,000 data points, obtained by sampling 1,000 points from each of 10 Gaussian distributions with identical standard deviations but distinct means. This setup yields ten equally sized classes with similar distribution, designed to emphasize disproportionate token allocation and make token representation shrinkage patterns more easily observable. To investigate token representation shrinkage behavior under varying data complexity, we generate synthetic datasets with different input dimensionalities. And to further validate existence of token representation shrinkage, we adopt CIFAR-10 to do conduct experiments.

For experiments regarding generative model, we adopt ImageNet-100 which is a subset of the ImageNet-1K dataset containing 100 classes. The original ImageNet-100 comprises approximately 130,000 training images and 5,000 test images. To better evaluate both reconstruction-FID (r-FID) and generation-FID (g-FID), we uniformly sampled total 20,000 images from all training classes to build up test dataset and construct an additional validation set containing 5,000 images. For the medical domain, we adopt the Ocular Disease Recognition (ODIR) [21] dataset, which contains 6,716 fundus images labeled across 8 diagnostic categories. We using a 70%/20%/10% split to partition the data into training, test, and validation sets.

Metrics For the synthetic dataset, we directly visualize the original data and its reconstructions, along with the corresponding token and embedding distributions, as shown in Fig. 3. For high-dimensional data, t-SNE is applied for dimensionality reduction prior to visualization.

To quantify the token representation shrinkage problem, we utilize cosine distance and perplexity. The average pairwise cosine distance across the codebook serves as an indicator of code clustering, with lower values suggesting that the code vectors have concentrated in a limited angular region. The perplexity, which is computed by the entropy over the codebook likelihood, reflects the effective tokens being utilized and is maximized when all tokens are used uniformly.

To evaluate the tokenizer's reconstruction performance, we adopt reconstruction FID (r-FID), mean squared error (MSE), and LPIPS scores. For generative quality, we utilize generation FID (g-FID) as the primary metric. To assess the diversity and distributional coverage of generated samples, we compute the average pairwise pixel-level distance between generated images.

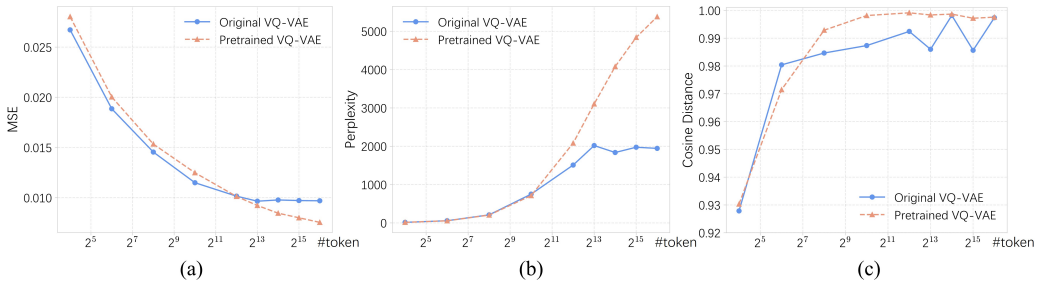


Figure 4: **Validation of token representation shrinkage on CIFAR-10** (a) Token representation shrinkage lead high reconstruction errors. (b) Token representation shrinkage leads to lower perplexity, demonstrating lower token utilization. (c) Token representation shrinkage leads to higher similarity among token during training.

Training Configuration For generative model experiments, we follow the tokenizer framework proposed in VQGAN [10]. Due to resource limitation, we resize all input images to 128×128 resolution and reduce the backbone's channel size to 64. To preserve a 16×16 latent spatial resolution, one downsampling layer and its upsampling layer are removed. All tokenizer experiments are conducted with a fixed codebook size of 16,384. To ensure feasibility under limited resources, we use the smallest generative model configurations. The MaskGIT generator employs a ViT [9] with depth of 24, while the VAR model uses a depth of 16. All tokenizers and generative models are trained on 2 A100 GPUs with 40 GB memory. Training the tokenizers on ImageNet-100 typically takes 1.5 to 3 days, while training the generative models requires 3-6 days depends on setting. Complete training details and hyperparameters are provided in the supplementary material.

5.2 CIFAR-10 Results

To validate that the shrinkage exists under real-world data conditions, we conducted corresponding experiments on the CIFAR-10 dataset. Additionally, we hypothesize that given a fixed dimensionality of the representation space, an increase in the number of tokens tends to facilitate their clustering, thereby making token representation shrinkage more pronounced. Under these conditions, the disadvantages caused by token representation shrinkage problem likely become more evident. Therefore, we evaluated the performance VQVAE's performance across varying token quantities.

As shown in Fig. 4, the original VQ model performs well when the number of tokens is relatively small. However, as the token count increases, particularly beyond 2^{12}, its reconstruction performance deteriorates relative to the pretrained counterpart. Notably, the perplexity curve of the original VQ flattens after 2^{13} tokens, indicating poor token utilization. Additionally, its average cosine distance remains consistently lower than that of the pretrained model, suggesting a higher degree of similarity among tokens. These findings collectively indicate that the token shrinkage problem becomes increasingly severe as the token set grows, leading to reduced representational diversity.

One possible reason about pretrained method underperforms the original approach at low token numbers is the gap between the discrete representations learned during pretraining and the continuous representations during finetuning, which poses challenges to the VQ learning process. However, this negative impact is outweighed by the benefits of our solution as the codebook size increases. Overall, our approach not only addresses token representation shrinkage but also unleashes the potential of VQ, further leveraging the benefits of a large codebook. Additionally, exploring how to mitigate the performance gap when the token number is low remains a worthy avenue for further investigation.

5.3 ImageNet-100 Results

Tokenizer performance Both types of original tokenizers exhibit a clear token representation shrinkage problem as shown in Tab. 1. For the tokenizer used in MaskGIT [4], we observe limited variation among tokens indicated by relatively small cosine distances (0.67 *vs.* 0.94). It reflects the high similarity between tokens. In addition, the tokenizer exhibits low perplexity (924.57 *vs.* 5311.88), suggesting that only a small subset of tokens is frequently utilized. Together, these observations

Table 1: **Performance evaluation of various tokenizers on the ImageNet-100 dataset.** "Shrink" indicates whether token representation shrinkage is present (✓) or mitigated using our proposed method (✗).

Tokenizer	Shrink	r-FID ↓	MSE ↓	LPIPS ↓	Cosine. ↑	Perp. ↑
MaskGIT	✗	**8.58**	**3.28**	**2.34**	**0.94**	**5311.88**
	✓	12.22	3.91	2.70	0.67	924.57
VAR	✗	**5.04**	**2.22**	**1.63**	**0.97**	**7044.51**
	✓	5.39	2.60	1.85	0.64	2801.88

Table 2: **ImageNet-100 generation**

Model	Shrink	g-FID ↓	Pixel Dist. ↑
MaskGIT	✗	**14.60**	**80.77**
	✓	14.75	75.89
VAR	✗	**10.70**	**75.92**
	✓	12.88	70.69

Table 3: **ODIR generation**

Model	Shrink	g-FID ↓	Pixel Dist. ↑
VAR	✗	**34.33**	**49.83**
	✓	37.65	49.01

imply that token usage is poorly aligned with the embedding space, pointing to a clear case of token representation shrinkage. However, after pretraining, tokens are more evenly utilized and better aligned with the embedding space. These observations confirm that pretraining effectively mitigates the token representation shrinkage phenomenon. As a result, the pretrained tokenizer achieves improved reconstruction performance, with lower r-FID (8.58 *vs.* 12.22), LPIPS (2.34 *vs.* 2.70), and MSE (3.28 *vs.* 3.91).

A similar pattern is also observed for the multi-scale tokenizer in VAR. Without pretraining, severe token representation shrinkage is evident. Pretraining once again proves effective in alleviating this issue, leading to more balanced token usage and enhanced reconstruction performance.

Generative Performance Token representation shrinkage significantly impairs the creativity of generative models, manifesting as a decline in both image quality and diversity as shown in Tab. 2. For the MaskGIT model, we observe that token representation shrinkage leads to a noticeable degradation in the generation FID (g-FID), indicating a reduction in the visual fidelity of synthesized images. Additionally, the pairwise pixel distance among generated samples is substantially reduced, suggesting that some outputs are highly similar. This phenomenon reflects a collapse in output variation, which we attribute directly to the narrowing of the token distribution(token representation shrinkage).

For the VAR model, we also observe a loss of creativity resulting from token representation shrinkage. Without proper mitigation, shrinkage in its multi-scale tokenizer leads to reduced generation quality and a clear drop in diversity. These results reinforce the conclusion that inadequate token representation limits the model's ability to capture the full generative distribution, ultimately compromising its overall creativity. The generated images are shown in Fig. 5.

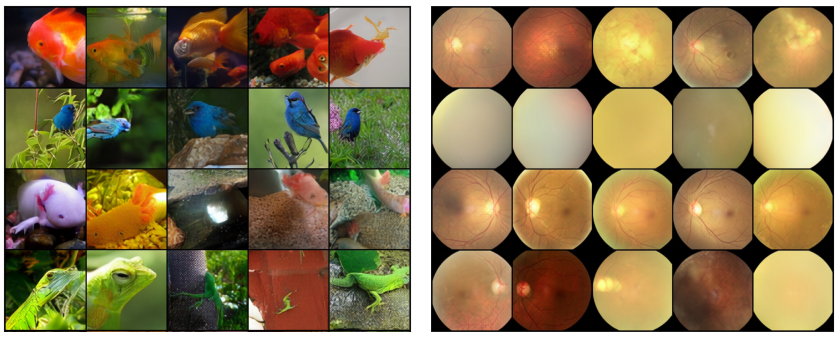
5.4 Real-world Medical Data Results

To further validate our findings, we conduct experiments across different image modalities within the ODIR medical image dataset. For the VAR model, we again confirm the presence of token representation shrinkage as shown in the Tab. 4. Additionally, we observe a corresponding decline in generative performance, including noticeable reductions in both image quality and diversity(Table), consistent with our observations on natural image datasets.

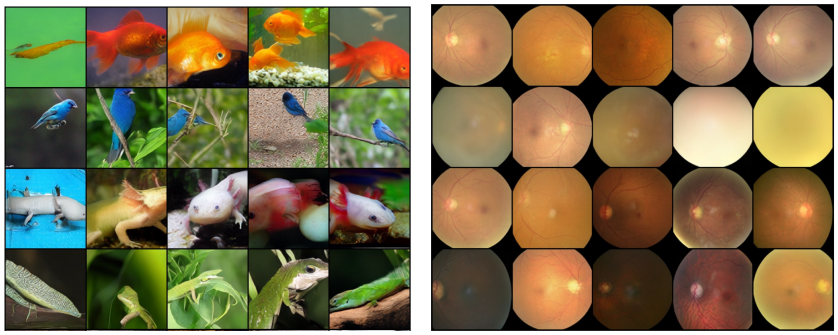
However, for the MaskGIT model, the results deviate from our expectations. Despite clear evidence of token representation shrinkage in the tokenizer, the generated images do not exhibit a drop in creativity. This suggests a decoupling between token representation shrinkage and generation degradation in this particular setting. We hypothesize that this discrepancy may be attributed to the relatively small dataset size and limited inherent diversity within the ODIR dataset, which potentially

Table 4: **Performance evaluation of VAR tokenizer on the ODIR medical dataset.**"Shrink" indicates whether token representation shrinkage is present (✓) or mitigated using our proposed method (✗).

Model	Shrink	r-FID ↓	MSE ↓	LPIPS ↓	Cosine. ↑	Perp. ↑
VAR	✗	11.04	2.05	6.79	0.90	5396.17
	✓	10.91	2.57	8.79	0.62	940.55



(a) w/ shrinkage



(b) w/o shrinkage

Figure 5: Generated images based on VAR. (a) ImageNet (a.left) and real-world medical images of eyes (a.right) generated using VAR as generative model and tokenizer **with token representation shrinkage**. (b) ImageNet (b.left) and real-world medical images of eyes (b.right) generated using VAR as generative model and tokenizer **without token representation shrinkage.**

masks the adverse effects of token representation. Detailed quantitative results are provided in the supplementary.

6 Conclusion

In this work, we systematically investigate the problem of token representation shrinkage in vector quantization, which is a critical yet overlooked factor contributing to mode collapse in transformer-based generative models. We demonstrate that commonly adopted token initialization strategies, especially those based on untrained encoders, lead to a collapse in token usage and embedding diversity, ultimately impairing the creativity of generative models by reducing output diversity and fidelity. To address this, we proposed a simple and effective two-stage training method that involves pretraining the encoder without VQ followed by fine-tuning with VQ. Our theoretical analysis and extensive experiments across synthetic, natural, and medical datasets confirm that this approach mitigates shrinkage, enhances token utilization, and improves generative performance. These findings highlight the importance of tokenizer design and initialization in discrete representation learning and open up new avenues for further research on improving generative expressiveness in VQ-based models.

References

[1] G. Baykal, M. Kandemir, and G. Unal. Edvae: Mitigating codebook collapse with evidential discrete variational autoencoders. *Pattern Recognition*, 2024.

[2] Y. Bengio, N. Léonard, and A. Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013.

[3] J. Betker, G. Goh, L. Jing, T. Brooks, J. Wang, L. Li, L. Ouyang, J. Zhuang, J. Lee, Y. Guo, et al. Improving image generation with better captions. *Computer Science. https://cdn. openai. com/papers/dall-e-3. pdf*, 2(3):8, 2023.

[4] H. Chang, H. Zhang, L. Jiang, C. Liu, and W. T. Freeman. Maskgit: Masked generative image transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11315–11325, 2022.

[5] M. Chen, A. Radford, R. Child, J. Wu, H. Jun, D. Luan, and I. Sutskever. Generative pretraining from pixels. In *International conference on machine learning*, pages 1691–1703. PMLR, 2020.

[6] T. Chen, L. Li, and Y. Sun. Differentiable product quantization for end-to-end embedding compression. In *International Conference on Machine Learning*, 2020.

[7] T. M. Cover. *Elements of information theory*. John Wiley & Sons, 1999.

[8] P. Dhariwal, H. Jun, C. Payne, J. W. Kim, A. Radford, and I. Sutskever. Jukebox: A generative model for music. *arXiv preprint arXiv:2005.00341*, 2020.

[9] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[10] P. Esser, R. Rombach, and B. Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021.

[11] T. Gautam, R. Pryzant, Z. Yang, C. Zhu, and S. Sojoudi. Soft convex quantization: Revisiting vector quantization with convex optimization. *arXiv preprint arXiv:2310.03004*, 2023.

[12] A. Gersho and R. M. Gray. *Vector quantization and signal compression*. Springer Science & Business Media, 2012.

[13] N. Goswami, Y. Mukuta, and T. Harada. Hypervq: Mlr-based vector quantization in hyperbolic space. *arXiv preprint arXiv:2403.13015*, 2024.

[14] K. Hsu, W. Dorrell, J. Whittington, J. Wu, and C. Finn. Disentanglement via latent quantization. *Advances in Neural Information Processing Systems*, 2024.

[15] I. Huijben, M. Douze, M. Muckley, R. Van Sloun, and J. Verbeek. Residual quantization with implicit neural codebooks. *arXiv preprint arXiv:2401.14732*, 2024.

[16] L. Kaiser, S. Bengio, A. Roy, A. Vaswani, N. Parmar, J. Uszkoreit, and N. Shazeer. Fast decoding in sequence models using discrete latent variables. In *International Conference on Machine Learning*, 2018.

[17] D.-K. Le Tan, H. Le, T. Hoang, T.-T. Do, and N.-M. Cheung. Deepvq: A deep network architecture for vector quantization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018.

[18] D. Lee, C. Kim, S. Kim, M. Cho, and W.-S. Han. Autoregressive image generation using residual quantization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[19] J. Macqueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability/University of California Press*, 1967.

[20] R. Mama, M. S. Tyndel, H. Kadhim, C. Clifford, and R. Thurairatnam. Nwt: towards natural audio-to-video generation with representation learning. *arXiv preprint arXiv:2106.04283*, 2021.

[21] A. Maranhão. Ocular disease intelligent recognition (odir). `https://www.kaggle.com/datasets/andrewmvd/ocular-disease-recognition-odir5k`, 2020.

[22] F. Mentzer, D. Minnen, E. Agustsson, and M. Tschannen. Finite scalar quantization: Vq-vae made simple. *arXiv preprint arXiv:2309.15505*, 2023.

[23] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, and I. Sutskever. Zero-shot text-to-image generation. In *International conference on machine learning*, 2021.

[24] W. Shin, G. Lee, J. Lee, E. Lyou, J. Lee, and E. Choi. Exploration into translation-equivariant image quantization. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023.

[25] K. Tian, Y. Jiang, Z. Yuan, B. Peng, and L. Wang. Visual autoregressive modeling: Scalable image generation via next-scale prediction. *Advances in neural information processing systems*, 37:84839–84865, 2024.

[26] A. Van Den Oord, O. Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 2017.

[27] A. Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.

[28] J. Yu, X. Li, J. Y. Koh, H. Zhang, R. Pang, J. Qin, A. Ku, Y. Xu, J. Baldridge, and Y. Wu. Vector-quantized image modeling with improved vqgan. *arXiv preprint arXiv:2110.04627*, 2021.

[29] L. Yu, Y. Cheng, K. Sohn, J. Lezama, H. Zhang, H. Chang, A. G. Hauptmann, M.-H. Yang, Y. Hao, I. Essa, et al. Magvit: Masked generative video transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[30] N. Zeghidour, A. Luebs, A. Omran, J. Skoglund, and M. Tagliasacchi. Soundstream: An end-to-end neural audio codec. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2021.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification:We conducted experiments both on synthetic and real-world datasets to demonstrate the existence of token shrinkage. Furthermore, we evaluate on different generative models across different datasets to show our main claim that token shrinkage will affect the creativity of generative model.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: While our solution effectively mitigates token representation shrinkage in most settings, we observe that its performance may be suboptimal when the codebook size is small. We provide a preliminary analysis of this behavior and suggest that addressing this limitation presents an interesting direction for future research.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: Our paper includes a theoretical formulation of the token representation shrinkage phenomenon. We provide a mathematical definition and accompanying derivations to characterize its impact on generative model performance. While our work does not include formal theorems or lemmas, the assumptions and reasoning are clearly presented to support the analysis.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We point out the key point of our experiments setting and specify how we process the dataset.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We will submit our experimental code as supplemental material. Meanwhile, we plan to release our code in github after adding necessary comments and guidelines.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: In main text, we provide all necessary details about our modification to models. Additional implementation details and comprehensive training configurations are provided in the supplementary material.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

Justification: Although we did not report error bars, confidence intervals, or statistical significance tests, we conduct experiments on Cifar-10 across different number of tokens to validate our hypothesis. Further we conduct experiments across dataset and model to validate the our token shrinkage will impair creativity of generative model.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We point the most resource-intensive part of our experiments lies in generative modeling. All experiments for ImageNet-100 were conducted on A100 GPU (40GB), with tokenizer training taking approximately 1.5–3 days and VAR/MaskGIT training requiring 3–6 days.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We confirm that the research presented in this paper fully complies with the NeurIPS Code of Ethics in all respects. We also affirm that anonymity has been properly preserved.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.

- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: By identifying fundamental limitations such as token representation shrinkage, our study provides new insights that can guide future improvements in model design, training strategies, and evaluation methods. Ultimately, enhancing the creativity of generative models will broaden their applicability across domains such as art, design, healthcare, and scientific discovery.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: Our paper does not release any new models or datasets and only uses publicly available resources such as ImageNet, CIFAR-10, ODIR, as well as the MaskGIT and VAR models. These assets are well-established in the research community and are not known to pose a high risk of misuse. Therefore, no additional safeguards are necessary.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.

- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We used publicly available datasets including ImageNet, CIFAR-10, and ODIR, as well as the MaskGIT and VAR model, all of which are properly licensed and cited. For ImageNet and CIFAR-10, we adhere to their original licenses and usage terms as specified by their maintainers. The ODIR dataset is obtained from Kaggle and used in accordance with its published license and terms of service. The VAR and MaskGIT models are used as released by the authors, and we cite the original paper in our submission. We do not modify any of these assets and ensure compliance with their respective licenses.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We include our code in the supplementary materials for reproducibility and will release the complete project code publicly upon publication.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.