Improved Masked Image Generation with Token-Critic

José Lezama, Huiwen Chang, Lu Jiang, and Irfan Essa

Google Research

Abstract. Non-autoregressive generative transformers recently demonstrated impressive image generation performance, and orders of magnitude faster sampling than their autoregressive counterparts. However, optimal parallel sampling from the true joint distribution of visual tokens remains an open challenge. In this paper we introduce Token-Critic, an auxiliary model to guide the sampling of a non-autoregressive generative transformer. Given a masked-and-reconstructed real image, the Token-Critic model is trained to distinguish which visual tokens belong to the original image and which were sampled by the generative transformer. During non-autoregressive iterative sampling, Token-Critic is used to select which tokens to accept and which to reject and resample. Coupled with Token-Critic, a state-of-the-art generative transformer significantly improves its performance, and outperforms recent diffusion models and GANs in terms of the trade-off between generated image quality and diversity, in the challenging class-conditional ImageNet generation.

Keywords: generative models, vision transformer, diffusion process, image generation

1 Introduction

Class-conditional image synthesis is a challenging task, requiring the generation of varied and semantically meaningful images with realistic details and few or none visual artifacts. The field has seen impressive progress in the hand of mainly three techniques: large Generative Adversarial Networks (GANs) [3], diffusion models [9, 19], and transformer-based models over a vector-quantized (VQ) latent space [12, 5]. Each of these techniques presents different advantages trading-off model size, computational cost of sampling, image quality and diversity.

Building upon the transformers [39] for the natural language generation tasks [4], generative vision transformers achieved impressive image generation performance. While early works applied an autoregressive transformer in the VQ latent space [29, 12], recently the state-of-the-art on the common ImageNet benchmark was further advanced by a new model called MaskGIT [5] using mask-and-predict training inspired by BERT [8] and non-autoregressive sampling adapted from neural machine translation [13, 26].

To be more specific, during inference, MaskGIT [5] starts from a blank canvas with all the tokens masked out. In each step, it predicts all tokens in parallel but

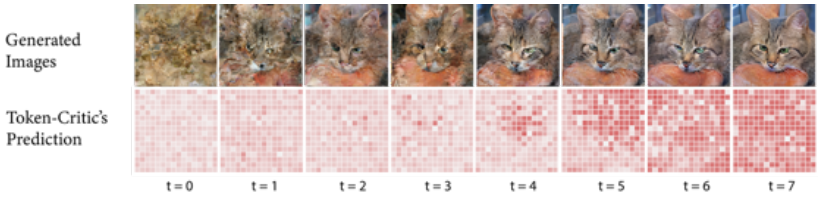


Fig. 1: Overview of the sampling procedure using Token-Critic. At each sampling iteration, Token-Critic predicts a high score for the tokens that are more likely sampled together under the joint distribution. Tokens with lower score are masked and resampled at the next iteration.

only keeps the ones with the highest prediction scores. The remaining tokens are masked out and will be re-predicted (resampled) in the next iteration until all tokens are generated with a few iterations of refinement. The non-autoregressive nature of MaskGIT allows orders-of-magnitude faster sampling, generating an image typically in 8-16 steps as opposed to hundreds of steps in autoregressive transformers [12] and diffusion models [9, 19].

One of the central challenges of iterative non-autoregressive generation is knowing how many and which tokens to keep and which to resample at each sampling step. For instance, MaskGIT [5] uses a predefined masking schedule and keeps the predicted tokens for which the model's prediction is more confident. However, this procedure presents three notable drawbacks. First, to select tokens to resample, it relies on the generator's predicted confidences which can be sensitive to modeling errors. Secondly, the decision to reject or accept is made independently for each token, which impedes capturing rich correlations between tokens. In addition, the sampling procedure is greedy and "non-regrettable", which does not allow to correct previously sampled tokens, even if they become less likely given the latest context.

In this work, we propose *Token-Critic*, a second transformer that takes as input the output of the generative transformer (or generator for short). Intuitively, the Token-Critic is trained to recognize configurations of tokens likely under the real distribution, and those that were sampled from the generator. During the iterative sampling process, the scores predicted by Token-Critic are used to select which token predictions are kept, and which are masked and resampled in the next iteration (*c.f.* Fig. 1).

With Token-Critic we tackle the three aforementioned limitations: 1) the masking of tokens is delegated to the Token-Critic model, trained to distinguish which tokens are unlikely under the true distribution. 2) Token-Critic looks at the entire set of sampled tokens collectively, thus is capable of capturing (spatial or semantic) correlations between tokens. 3) The proposed sampling procedure allows to correct previously sampled tokens during the iterative decoding.

When using Token-Critic, the state-of-the-art non-autoregressive generative transformer MaskGIT [5] significantly improves its performance on ImageNet 256×256 and 512×512 class-conditional generation, while achieving a better

trade-off between image quality and diversity. Furthermore, the gain obtained by using Token-Critic is complementary to the gain obtain by a pretrained ResNet classifier for rejection sampling. When coupled with classifier-based rejection sampling [31], Token-Critic parallels or surpasses the state-of-the-art continuous diffusion models with classifier guidance [9] in image synthesis quality while offering two orders of magnitude faster in generating images during inference.

2 Background

2.1 Non-autoregressive Generative Image Transformer

Generally, transformer-based models generate images in two stages [30, 12]. First, the image is quantized into a grid of discrete tokens by a Vector-Quantized (VQ) autoencoder built upon VAE [29, 31], GAN [12], or vision transformer backbones [40], in which each token is represented as an integer index in a codebook. In the second stage, an autoregressive transformer decoder [6] is learned on the flattened token sequence to generate image tokens sequentially based on the previously generated result (*i.e.* autoregressive decoding). In the end, the generated codes are mapped to pixel space using the decoder obtained from the first stage.

Non-autoregressive transformers [13, 15, 26], which were originally proposed for machine translation, are, very recently, extended to improve the second stage of autoregressive decoding [25]. For example, MaskGIT [5] demonstrates highly-competitive fidelity and diversity of conditional image synthesis on the ImageNet benchmark as well as faster inference than the autoregressive transformer [12] in addition to the diffusion models [9, 28]. To be specific, MaskGIT is trained on the masked language modeling (MLM) proxy task proposed in BERT [8]. During inference, the model adopts a non-autoregressive decoding method to synthesize an image in a constant number of steps (typically 8-16 steps) [15]. Starting with all the tokens masked out, in each inference step, MaskGIT predicts all tokens simultaneously in parallel and only keeps the ones with the highest prediction scores. The remaining tokens are masked out and will be re-predicted in the next iteration. The mask ratio is made decreasing, according to a cosine function, until all tokens are generated. In the following, Sections 2.2 and 2.3 describe limitations in the training and sampling of the MaskGIT model. Then, in Section 3 we introduce the Token-Critic as a solution to mitigate these limitations.

2.2 Challenges in Training Non-Autoregressive Transformers

Ideally, one would like the masked generative transformer to learn the joint distribution of unobserved tokens $\mathbf{x} = [x_1, \dots x_N]$ given the observed tokens $\mathbf{o}$. Both $\mathbf{x}$ and $\mathbf{o}$ are sequences of N tokens where N (*e.g.*, 16×16) indicates the latent size of the VQ autoencoder obtained in the first stage. Each $x_j \in \mathcal{V} = \{1, \cdots, K\}$ is an integer token in the codebook of size K. Notice the element in $\mathbf{o}$ can take the value of a special mask token, *i.e.* $o_j \in \mathcal{V} \cup \{\texttt{[MASK]}\}$. We shall refer to their true joint distribution as $q(x_1, \dots, x_N|\mathbf{o})$.

Current non-autoregressive generative transformers [5, 13] are trained to optimize the sum of *the marginal* cross-entropies for each unobserved token:

$$\mathcal{L}_i = -\sum_{j=1}^{N}\sum_{k=1}^{K} \tilde{q}(x_j = k|\mathbf{o}) \log p_\theta(x_j = k|\mathbf{o}), \tag{1}$$

where $\tilde{q}$ represents an approximation to the true marginal given by considering one random real sample.

A limitation is that optimizing over the marginals hinders capturing the richness of the underlying joint distribution of unobserved tokens. Essentially, this training scheme is equivalent to minimizing the Kullback-Leibler (KL) divergence between the data and model distributions, both approximated as fully factorizable distributions.

2.3 Challenges in Sampling from Non-autoregressive Transformers

During sampling, one is interested in sampling from the full joint distribution of unobserved tokens $q(x_1, \dots x_N|\mathbf{o})$. However, even if the transformer representations are distributed, the output for each token models its sampling distribution independently. More precisely, for a given unobserved token x_t, a value is sampled from $p_\theta\left(x_t|\mathbf{z}, \mathbf{o}\right) = p_\theta\left(x_t|\mathbf{z}\right)$, where $\mathbf{z}$ is the latent embedding visible by all output tokens (*i.e.* the activations of the last attention layer). Sampling from the true distribution would require coordinating the values of all sampled tokens, which is not possible with the current architecture (unless the sampling is made deterministic, which harms the diversity of the generated images). Thus, non-autoregressive vision transformers still need to resort to iterative ancestor sampling. Typically, in each step of the sampling process, a growing subset of the tokens is accepted and the rest is rejected and resampled.

Aiming at better approximating the true joint distribution, the question of how to select which sampled tokens to keep and which to resample is the main focus of this work. We propose to do this using an auxiliary model that we term the Token-Critic. The Token-Critic is a second transformer trained to individually identify which tokens in a sampled vector-quantized image are plausible under the true joint distribution and which are not. During the iterative non-autoregressive sampling procedure, the Token-Critic is used in each iteration to reject the tokens that are less likely given the context.

3 Method

The goal of Token-Critic is to guide the iterative sampling process of a non-autoregressive transformer-based generator. Given the tokenized image outputted by the generative transformer, Token-Critic is designed as a second transformer that provides a score for each token, indicating whether the token is likely under the real distribution, given its context.

In Section 3.1, we first introduce the procedure for training the Token-Critic and in Section 3.2 we describe how it is used during sampling. At all times we assume a pre-trained non-autoregressive transformer generator is available. Finally, in Section 3.3 we explain the role of the Token-Critic by drawing a connection to discrete diffusion processes.

3.1 Training the Token-Critic

The training procedure for Token-Critic is straightforward. Given a masked image and its corresponding completion by the generative non-autoregressive transformer, the Token-Critic is trained to distinguish which of the tokens in the resulting image were originally masked.

More specifically, consider a real vector-quantized image $\mathbf{x}_0$, a random binary mask $\mathbf{m}_t$ and the resulting masked image $\mathbf{x}_t = \mathbf{x}_0 \odot \mathbf{m}_t$. The subindex t indicates the masking ratio, as will be detailed shortly. First, the generative transformer G_θ, parameterized by θ, is used to predict the masked tokens, namely sampling $\tilde{\mathbf{x}}_0$ from $p_\theta(\hat{\mathbf{x}}_0|\mathbf{x}_t, c)$, in which to condition on the class index c, we prepend a class token to the flattened set of visual tokens. The unmasked tokens in $\mathbf{x}_t$ are copied into the output to form $\hat{\mathbf{x}}_0 = \tilde{\mathbf{x}}_0 \odot \mathbf{m}_t + \mathbf{x}_0 \odot (1 - \mathbf{m}_t)$.

The Token-Critic transformer, parameterized by ϕ, takes as input $\hat{\mathbf{x}}_0$ and outputs a predicted binary mask for $\mathbf{m}_t$. During training, the parameters ϕ are optimized to minimize the following objective:

$$\mathcal{L}_i = \mathop{\mathbb{E}}_{q(\mathbf{x}_0,c)q(t)q(\mathbf{m}_t|t)p_\theta(\hat{\mathbf{x}}_0|\mathbf{m}_t\odot\mathbf{x}_0,c)} \Big[\sum_{j=1}^{N} BCE\Big(\mathbf{m}_t^{(j)},\ p_\phi(\mathbf{m}_t^{(j)}|\hat{\mathbf{x}}_0, c)\Big)\Big], \qquad (2)$$

where $q(\mathbf{x}_0, c)$, $q(t)$, $q(\mathbf{m}_t|t)$ are the distributions of real unmasked images, timesteps, and binary masks, respectively, and BCE denotes the binary cross-entropy loss. The sampling distribution $p_\theta(\hat{\mathbf{x}}_0|\mathbf{m}_t \odot \mathbf{x}_0)$ induced by the generator G_θ is held fixed during the training of the Token-Critic model.

The training algorithm is summarized as pseudocode in Algorithm 1. Notice that $\gamma(t) \in (0, 1)$ in Step 4 is the cosine mask scheduling function. Given a uniform random number t sampled from $q(t) = \mathcal{U}(0, 1)$, the number of masked tokens in $\mathbf{m}_t$ is computed as $r = \lceil N \cdot \gamma(t) \rceil$, where N is the total number of tokens within an image.

3.2 Sampling with Token-Critic

During inference, we are interested in progressively replacing masked tokens with an actual code in the vocabulary. Starting from a fully masked image $\mathbf{x}_T$ and the class condition c, we iteratively sample from $p(\mathbf{x}_{t-1}|\mathbf{x}_t, c)$, which may be approximated by:

$$p(\mathbf{x}_{t-1}|\mathbf{x}_t, c) = \sum_{\mathbf{x}_0} p_\phi(\mathbf{x}_{t-1}|\mathbf{x}_0, c) p_\theta(\mathbf{x}_0|\mathbf{x}_t, c) \tag{3}$$

$$= \mathop{\mathbb{E}}_{\mathbf{x}_0 \sim p_\theta(\mathbf{x}_0|\mathbf{x}_t, c)} \Big[p_\phi(\mathbf{x}_{t-1}|\mathbf{x}_0, c) \Big] \tag{4}$$

$$\approx p_\phi(\mathbf{x}_{t-1}|\hat{\mathbf{x}}_0, c),\ \hat{\mathbf{x}}_0 \sim p_\theta(\mathbf{x}_0|\mathbf{x}_t, c). \tag{5}$$

In (4), we assume that $\mathbf{x}_{t-1}$ is conditionally independent of $\mathbf{x}_t$ given $\mathbf{x}_0$. We will get back to this assumption shortly. In (5), the expectation is empirically approximated using a single sample Monte Carlo for $p_\theta(\mathbf{x}_0|\mathbf{x}_t, c)$, which is obtained from the output of the generative transformer G_θ. The next step is to sample from $p_\phi(\mathbf{x}_{t-1}|\hat{\mathbf{x}}_0, c)$. Recall that $\mathbf{x}_{t-1}$ is a masked version of $\hat{\mathbf{x}}_0$, rendering it solely determined by $\hat{\mathbf{x}}_0$ and a mask $\mathbf{m}_{t-1}$. Thus, we can sample $\mathbf{x}_{t-1}$ in (5) using Token-Critic to predict a mask $\mathbf{m}_{t-1}$ given $\hat{\mathbf{x}}_0$.

Note that the mask computation of MaskGIT [5] only relies on the prediction score $p_\theta(\mathbf{x}_0|\mathbf{x}_t, c)$, in which tokens with the lowest predictions are masked. The mask sampling is independent for each token and moreover greedy which means previously unmasked tokens will be kept unmasked forever. In contrast, the proposed mask sampling is learned by the Token-Critic model ϕ to approximate sampling from the joint distribution by taking into account the correlation among tokens. This notably improves the sampling leading to better generation quality. Secondly, Token-Critic makes generation regrettable, allowing to revoke prior decisions based on the most recent generation.

The sampling process is given as pseudocode in Algorithm 2 and represented schematically in Figure 1. The rate of masking in each step is given by the scheduling function $\gamma(t)$, with $t = T-1 \ldots 0$, where higher values of t correspond to more masking. After predicting $\mathbf{m}_t$ in each step, we mask the $R = \lceil \gamma(t/T) \cdot N \rceil$ tokens with the lowest Token-Critic score. Following [5], to introduce randomness in the first steps, we add a small "selection noise" $\mathbf{n}(t)$ to the Token-Critic scores before ranking them. This selection noise is annealed according to $\mathbf{n}(t) = K \cdot \mathbf{u} \cdot (t/T)$, where K is a hyperparameter and $\mathbf{u} \in [-0.5, 0.5]^N$ is a random uniform vector. Furthermore, the sampling temperature for each token is also annealed according to a linear schedule $T(t) = a \cdot (t/T) + b$.

Finally, we get back to the assumption in (4) that $\mathbf{x}_{t-1}$ is made independent of $\mathbf{x}_t$ given $\mathbf{x}_0$. This assumption can be dropped by simply adapting the Token-Critic's input by concatenating the previous mask $\mathbf{m}_t$ to $\hat{\mathbf{x}}_0$. However, in practice, we find this does not yield a better result. In fact, by ignoring the previous mask, Token-Critic has the ability to correct previously sampled tokens that are no longer as likely given the latest context, which addresses the greedy mask selection in the MaskGIT model [5].

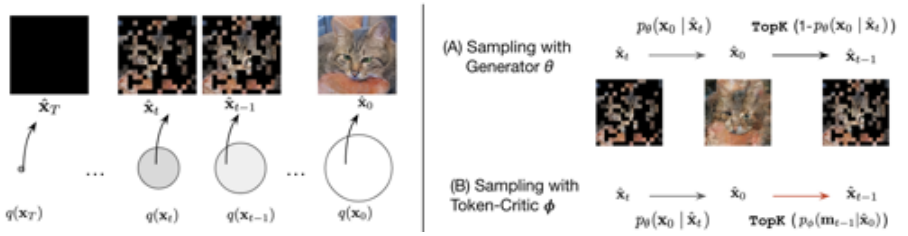


Fig. 2: Token-Critic in the lens of a discrete stochastic process that gradually masks a real image $\mathbf{x}_0$ from $t = 0 \dots T$, where $\mathbf{x}_T$ is fully masked. The circles represent the distributions of real vector-quantized images under different masking rates. During the reverse process, a masked image estimate $\hat{\mathbf{x}}_t$ is refined by first using the generator to predict a clean image $\hat{\mathbf{x}}_0$, and then predicting the mask for the next timestep. While prior works (A) use the confidence of the generator G_θ, we use the predictions of Token-Critic (B) to select which tokens to mask.

Algorithm 1 Token-Critic Training

Input: Pre-trained generator G_θ, scheduling function $\gamma(t)$, learning rate η
Output: Token-Critic parameters ϕ

repeat
  $\mathbf{x}_0, c \leftarrow$ i.i.d. sampled VQ image
  $t \sim \mathcal{U}_{(0,1)}$
  $\mathbf{m}_t \leftarrow$ random mask($\lceil \gamma(t) \cdot N \rceil$)
  $\mathbf{x}_t \leftarrow \mathbf{x}_0 \odot \mathbf{m}_t$
  $\hat{\mathbf{x}}_0 \leftarrow G_\theta(\mathbf{x}_t, c)$
  $\phi \leftarrow \phi - \eta \nabla_\phi BCE(\mathbf{m}_t, p_\phi(\mathbf{m}_t | \hat{\mathbf{x}}_0, c))$
until convergence

Algorithm 2 Token-Critic Sampling

$\mathbf{x}_T \leftarrow [\texttt{[MASK]}]_N$
for $t = T \dots 1$ **do**
  $k = \lceil \gamma((t-1)/T) \cdot N \rceil$
  $\hat{\mathbf{x}}_0 = G_\theta(\mathbf{x}_t, c)$
  $\{p_i\}_{i=1\dots,N} \leftarrow p_\phi(\mathbf{m}_{t-1}^{(i)} | \hat{\mathbf{x}}_0, c) + n(t)$
  $\tau \leftarrow$ rank$_k(\{p_i\})$
  $\{\mathbf{m}_{t-1}^{(i)}\} \leftarrow 1$ if $p_i > \tau$, else 0
  $\mathbf{x}_{t-1} = \hat{\mathbf{x}}_0 \odot \mathbf{m}_{t-1}$
end for

3.3 Relation to Discrete Diffusion Processes

The role of Token-Critic can also be understood under the perspective of discrete diffusion processes [1, 11, 16, 20], where it is assumed that there exists a stochastic process that gradually destroys information by masking. In this setting, the reverse process aims to progressively replace masked tokens with elements from the VQ codebook following the true distribution. In our case, this is what the generator transformer G_θ does in each step of the sampling procedure. Ideally, each intermediate result should lie within the distribution of partially masked real images, since this is the distribution used to train G_θ. The role of Token-Critic is to guide the intermediate samples towards these regions.

Figure 2 represents a schematic representation of the reverse sampling process. Given a current estimate of a masked image $\hat{\mathbf{x}}_t$, we use the generator to

produce an estimate clean image $\hat{\mathbf{x}}_0$. Note that due to the aforementioned modeling limitations, this estimate typically falls far from the distribution of real images. Token-Critic is then used to predict a less corrupted image $\hat{\mathbf{x}}_{t-1}$ from $\hat{\mathbf{x}}_0$. Since it was trained to distinguish incompatible tokens, the improved prediction is achieved by masking the least "plausible-looking" tokens.

In the diffusion processes literature, a similar sampling strategy relying on an estimate of the clean image was used in [37] for the continuous case and [20, 1, 16] for the discrete case. The difference in our approach is that we implicitly use a learned forward model instead of a fixed one obtained beforehand (*e.g.*, the Gaussian prior). On the other hand, previous discrete diffusion models for image generation [1, 11, 16, 20] typically assume a stochastic process that is independent for each token, and give a fixed form Markov chain that defines the probabilities of each token being masked, converted randomly or staying the same. Even under the independence assumption, if the number of token categories is large, the computation of the n-step Markov transition matrix required to obtain the posterior can be impractical. These design differences in part explain the diffusion models' low-efficiency when synthesizing high-resolution images. Instead, Token-Critic trades-off the analytical interpretability and tractability of these assumptions for a more efficient, learned forward process $p_\phi(\mathbf{x}_t|\hat{\mathbf{x}}_0)$.

Finally, we can derive the training objective of Token-Critic from the KL divergence between the distributions of real partially masked images $q(\mathbf{x}_{t-1})$, and the distribution of partially masked images obtained in the intermediate steps by the proposed sampling scheme $p_{\theta,\phi}(\mathbf{x}_t)$. We refer to the appendix for the derivation.

4 Experiments

In this section we evaluate the proposed approach on class-conditional image generation tasks on ImageNet [7] 256×256 and 512×512. We compare over classical metrics to examine the trade-off between quality and variablity, notably FID [18] vs. Inception Score [33] and Precision vs. Recall [27]. We observed that the highly-competitive baseline is significantly improved when using Token-Critic, and that the proposed method obtains an advantageous quality-diversity trade-off, compared to state-of-the-art GANs and continuous diffusion models.

4.1 Experimental Setup

We use a pre-trained MaskGIT [5] model as the generator, and use the Token-Critic to guide the sampling, as described in Section 3.2. We adopt the VQ encoder-decoder of [12] and [5], with a codebook with 1024 tokens, trained at 256×256 resolution in the same datasets. The VQ encoding compresses the image by a factor of 16, so that a 256×256 (512×512) image is represented as a grid of 16×16 (32×32) integers. The generator is a transformer with 24 layers and 16 heads. For the Token-Critic, we use a relatively smaller transformer

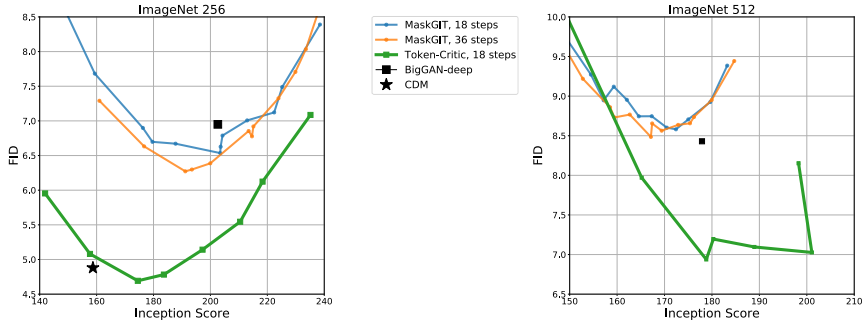


Fig. 3: FID-vs-IS curves on ImageNet 256x256 and 512x512 (bottom right is better). The trade-off between diversity and quality is traversed by varying the sampling temperature. Compared to the baseline [5], sampling using Token-Critic produces a significant improvement in performance, and outperforms BigGAN-deep [3] and CDM [19], achieving a new state-of-the-art for methods that do not rely on an external classifier.

with 20 layers and 12 heads, but otherwise of identical architecture. Both transformers use embeddings of dimension 768 and a hidden dimension of 3,072, learnable positional embedding [39], LayerNorm [2], and truncated normal initialization (stddev=0.02). The following training hyperparameters were used for both MaskGIT and Token-Critic: dropout rate=0.1, Adam optimizer [22] with $\beta_1 = 0.9$ and $\beta_2 = 0.96$. We use RandomResizeAndCrop for data augmentation. All models are trained on 8×8 TPU devices with a batch size of 256. The MaskGIT generators and the Token-Critic models were trained for 600 epochs. We use the same cosine schedule for the masking rate as in [5], for both training and sampling. We use 18 steps when sampling with Token-Critic, as we found this gives the best results.

4.2 Class-Conditional Image Synthesis

Quantitative Results We evaluate our method on class-conditional synthesis using ImageNet. Our main results are summarized in Table 1. For a more comprehensive quantitative comparison, we compare Inception Score vs. FID in Figure 3. These represent the trade-off between image quality, associated to Inception Score, and diversity or coverage, associated to FID. To traverse the quality-diversity trade-off for the baseline and Token-Critic, we modify the sampling temperature and selection noise parameter K (Section 3.2). Higher selection noise and temperature produce higher variability but lower quality.

We compare the proposed approach to the MaskGIT baseline which uses the generator's prediction confidence to select which tokens to reject in each step of the iterative sampling. To account for the fact that the proposed method requires two forward passes for each sampling step, in Figure 3 we compare the proposed approach to the MaskGIT baseline for double the number of sampling

Model	steps	ImageNet 256x256				ImageNet 512x512			
		FID ↓	IS ↑	Prec	Rec	FID ↓	IS ↑	Prec	Rec
BigGAN-deep [3]	1	6.95	202.65	**0.86**	0.24	8.43	177.9	**0.85**	0.25
ADM [9]	250	10.94	101.0	0.69	**0.63**	23.24	58.06	0.73	**0.60**
CDM [19]	100	4.88	158.7	n/a	n/a	n/a	n/a	n/a	n/a
MaskGIT [5]	18	6.56	**203.6**	0.79	0.48	8.48	167.1	0.78	0.46
MaskGIT+Token-Critic	18(x2)	**4.69**	174.5	0.76	0.53	**6.80**	**182.1**	0.73	0.50

Table 1: Comparison between methods that do not leverage an external classifier. We report the sampling configurations that obtain the best FID score for each method, and refer to Figure 3 for the more comprehensive trade-off between FID and Inception Score. All methods are evaluated on ImageNet training set. Results for [3] are as reproduced by [34].

Model	ImageNet 256x256				ImageNet 512x512			
	FID ↓	IS ↑	Prec	Rec	FID ↓	IS ↑	Prec	Rec
ADM+Guid. [9]	4.59	186.7	0.82	0.52	7.72	172.7	**0.87**	0.42
ADM+Guid.+Upsamp. [9]	3.94	215.8	**0.83**	0.53	3.85	221.7	0.84	**0.53**
StyleGAN-XL ($\Psi = 1.0$) [34]	**3.26**	225.6	0.74	0.45	**3.58**	219.8	0.73	0.43
MaskGIT [5] (a.r. 20%)	4.70	266	0.80	0.48	5.13	250.7	0.79	0.47
[5]+Token-Critic (a.r. 20%)	3.75	**287.0**	0.75	**0.55**	4.03	**305.2**	0.73	0.50

Table 2: Comparison between methods that use an external classifier during training or sampling. We report the sampling configurations that obtain the best FID score for each method. We refer to Figure 4 to better appreciate the improvement obtained by Token-Critic in the trade-off between image quality and sample diversity. Sampling with Token-Critic and a classifier rejection scheme significantly improves the baseline and obtains the best Inception Score of all compared methods.

steps. We also compare to state-of-the-art GAN architectures BigGAN-deep [3], and continuous Cascaded Diffusion Model (CDM) [19] and Ablated Diffusion Model (ADM) [9] without external classifier guidance.

Of all the methods that do not rely on an external classifier, the proposed approach achieves the lowest FID, while providing an advantageous FID / Inception Score balance. Compared to the MaskGIT baseline, it achieves a significant improvement in terms of FID and Inception Score.

Leveraging an External Classifier Classifier-guidance is a commonly adopted technique in diffusion models to improve class-conditional generation [12, 31], consisting on using the gradient of an external classifier to improve the class score of the sampled image, and thus render it more semantically meaningful. We show that the improvement obtained by leveraging an external pre-trained classifier is independent of the improvement brought by the Token-Critic. More-

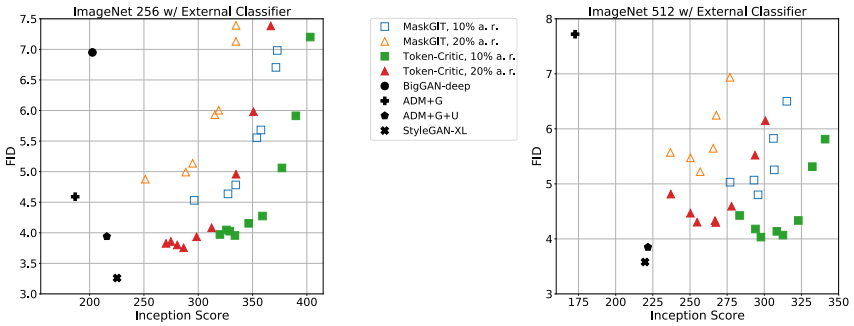


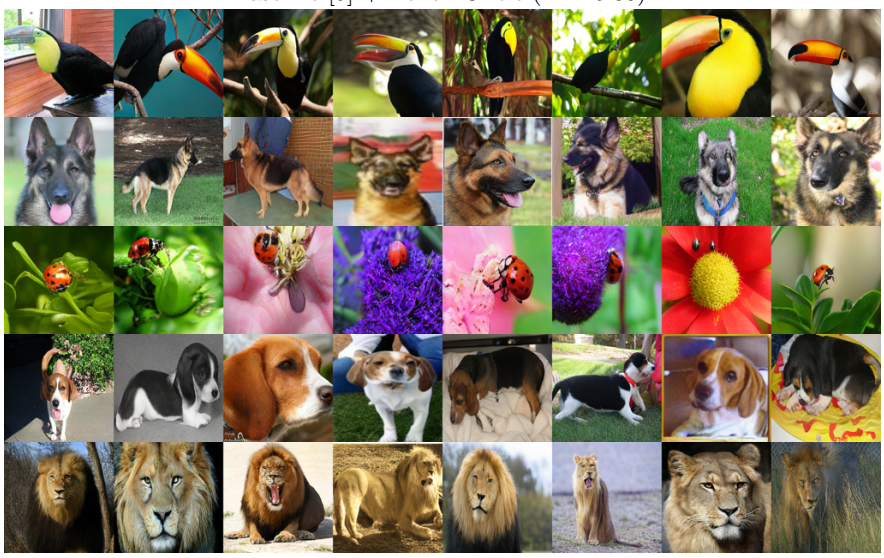
Fig. 4: FID-vs-Inception Score trade-off for methods that leverage an external classifier (bottom right is better). We use an external ResNet classifier for rejection sampling at acceptance rates 10% and 20%. Points in the graph indicate different sampling temperatures to balance quality and diversity, the remaining hyperparameters being equal. Token-Critic improves significantly upon [5], and also obtains a superior trade-off to diffusion models with upsampling [9]. The trade-off is also comparable to the concurrent work StyleGAN-XL [34], which obtains the best FID but with much lower Inception Score.

over, the combination of both further improves the quantitative performance, achieving the highest reported Inception Score, and FID scores competitive with the most advanced GANs and Diffusion Models that use an external classifier.

Since classifier guidance is not directly transferable to the VQ latent space, here we adopt a classifier-based rejection sampling scheme [31]. Given the conditioning class, we generate multiple image candidates and keep only the one with the highest classifier score for the class. For the external classifier we use a ResNet [17] with 50 layers. We experiment with acceptance rates of 10% and 20% (meaning we keep one out of 10 and one out of 5 images with the highest scores). Results are summarized in Table 2 and the FID vs. Inception Score curves are plotted in Figure 4. We include a comparison with concurrent work StyleGAN-XL [34]. Whilst StyleGAN-XL achieves better FID score, Token-Critic is superior with repsect to Inception Score, Precision and Recall.

Qualitative Results Figure 5 shows a qualitative comparison on ImageNet class-conditional generation between the baseline MaskGIT's original sampling procedure [5] and the proposed sampling using Token-Critic. We demonstrate the models without classifier-based rejection to better isolate the difference in image quality and diversity obtained by the proposed approach. Notably, sampling with the proposed approach achieves better structural consistency, showing the ability of the Token-Critic to capture long range dependencies. We refer to the supplementary material for further results and comparisons.

Baseline [5] + Token-Critic (FID 6.80).



Baseline [5] (FID 8.48).

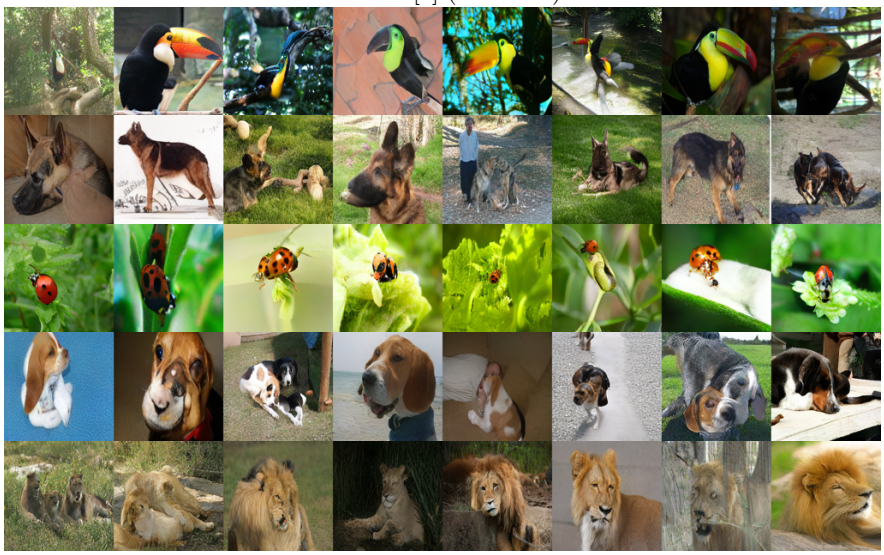


Fig. 5: Samples from ImageNet 512x512 models for classes Toucan (96), German Shepherd (235), Ladybug (301), Beagle (162) and Lion (291). All models ran for 18 steps.

Fig. 6: Refinement of previously generated vector-quantized images by [5] with an ImageNet 512x512 model. **Top:** Original samples (FID/IS 8.48/167). **Bottom:** After refining 60% of the tokens with lowest Token-Critic score (FID/IS 7.64/182.4). The semantics of the original image is maintained, but it is given a more realistic aspect.

4.3 VQ Image Refinement

To demonstrate the ability of Token-Critic to identify unlikely visual tokens, we apply it to refine the output of the baseline model. Given a generated VQ-encoded image by a MaskGIT generator, we compute the Token-Critic scores on the generated tokens, and proceed to resample the tokens that have low scores. We start by rejecting and replacing 60% of the original tokens in the first step, and then progressively reject and replace fewer tokens, following again a cosine schedule for $T = 9$ steps. The result of this procedure can be regarded as a visual quality improvement of the original images, see Figure 6. By applying this refinement procedure we improve the FID score of the baseline generator form 6.56 to 5.73 in 256×256 and from 8.48 to 7.64 in 512×512.

5 Related Work

While there exist other types of generative models such as VAEs [24, 38] and Flow-based models [32], we briefly review the works closely relevant to ours.
Generative Adversarial Networks (GANs) are capable of synthesizing high-fidelity images at blazing speeds. GAN based methods demonstrate impressive capability in yielding high-fidelity samples [14, 3, 21]. They suffer from, however, well known issues including training instability and mode collapse which causes a lack of sample diversity. Addressing these issues still remains an active research problem. Note that MaskGIT and Token-Critic are not affected by adversarial training instability, as the Token-Critic is trained asynchronously over a pre-trained MaskGIT.
Generative Image Transformers Inspired by the success of the transformer in the NLP field [8, 4], vision transformers [10] have been applied to various vision

tasks. In particular, the generative image transformer (GIT) [6] is inspired by the generative pre-trained transformer or GPT [4]. Generally, modern GITs consist of two stages [30]: image quantization and autoregressive decoding, where the former is to compress an image into tokens of a reasonable length whereas the latter, borrowed from neural machine translation [39], generate image tokens as if they were "visual words". Most recent contributions are on improving the first stage, *e.g.*, using vector-quantized models of various architectures and losses [12, 30, 40]. Very recently, [41, 5] proposed to use bi-directional transformers to synthesize images, which significantly accelerate the decoding time. Our work builds upon the MaskGIT model [5] and improves its mask sampling in non-autoregressive decoding.

Denoising Diffusion Models [36] define a parameterized Markov chain trained to reverse a forward process of corrupting a training image into pure noise. While many works have focused on continuous (Gaussian) diffusion processes [23], closely related to ours are diffusion models with *discrete* state spaces [35]. For example, Austin *et al.* [1] proposed a discrete diffusion (D3) model corrupting data by transition matrices that embed structure knowledge. Song *et al.* [37] introduced implicit diffusion models of non-Markovian diffusion processes. Hoogeboom *et al.* [20] modeled the categorical data through a fixed multinomial diffusion for image segmentation, which was improved in ImageBART [11] by combing with the autoregressive formulation.

The majority of diffusion models is characterized by a forward process with tractable known expressions according to [1], which is essential for permitting not only efficient forward sampling but also computation of the posterior. From this perspective, our method, similar to MaskGIT [5], is not a traditional diffusion model because it parameterizes the forward process by a transformer that does not have a tractable expression. Since the direct computation of the forward process is intractable in our case, we resort to learning a non-Markov transformer that can teleport to any forward state. This is achieved by the proposed second transformer (Token-Critic). Empirically, we found this strategy to be effective needing considerably fewer number of decoding steps (typically 8-16 steps) while producing competitive quality.

6 Conclusion

In this work, we proposed a novel method for sampling from a non-autoregressive generative vision transformer. It is based on using a second transformer, the Token-Critic, to select which tokens are accepted and which are rejected and resampled during the iterative generative process. Given a reconstructed masked image, the Token-Critic is trained to distinguish which visual tokens belong to the original image and which are predictions of the generative transformer. Coupled with the Token-Critic, an already powerful non-autoregressive transformer significantly improves its performance, and outperforms the state-of-the-art in terms of the trade-off between generated image quality and variety, in the challenging task of class-conditional ImageNet generation.

References

1. Austin, J., Johnson, D., Ho, J., Tarlow, D., van den Berg, R.: Structured denoising diffusion models in discrete state-spaces. Advances in Neural Information Processing Systems **34** (2021)
2. Ba, J.L., Kiros, J.R., Hinton, G.E.: Layer normalization. arXiv preprint arXiv:1607.06450 (2016)
3. Brock, A., Donahue, J., Simonyan, K.: Large scale gan training for high fidelity natural image synthesis. arXiv preprint arXiv:1809.11096 (2018)
4. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., Amodei, D.: Language models are few-shot learners. In: NeurIPS (2020)
5. Chang, H., Zhang, H., Jiang, L., Liu, C., Freeman, W.T.: Maskgit: Masked generative image transformer. arXiv preprint arXiv:2202.04200 (2022)
6. Chen, M., Radford, A., Child, R., Wu, J., Jun, H., Luan, D., Sutskever, I.: Generative pretraining from pixels. In: International Conference on Machine Learning. pp. 1691–1703. PMLR (2020)
7. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009)
8. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
9. Dhariwal, P., Nichol, A.: Diffusion models beat gans on image synthesis. Advances in Neural Information Processing Systems **34** (2021)
10. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: ICLR (2021)
11. Esser, P., Rombach, R., Blattmann, A., Ommer, B.: Imagebart: Bidirectional context with multinomial diffusion for autoregressive image synthesis. Advances in Neural Information Processing Systems **34** (2021)
12. Esser, P., Rombach, R., Ommer, B.: Taming transformers for high-resolution image synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12873–12883 (2021)
13. Ghazvininejad, M., Levy, O., Liu, Y., Zettlemoyer, L.: Mask-predict: Parallel decoding of conditional masked language models. In: EMNLP-IJCNLP (2019)
14. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: NeurIPS (2014)
15. Gu, J., Kong, X.: Fully non-autoregressive neural machine translation: Tricks of the trade. In: Findings of ACL-IJCNLP (2021)
16. Gu, S., Chen, D., Bao, J., Wen, F., Zhang, B., Chen, D., Yuan, L., Guo, B.: Vector quantized diffusion model for text-to-image synthesis. arXiv preprint arXiv:2111.14822 (2021)
17. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)

18. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. Advances in neural information processing systems **30** (2017)
19. Ho, J., Saharia, C., Chan, W., Fleet, D.J., Norouzi, M., Salimans, T.: Cascaded diffusion models for high fidelity image generation. Journal of Machine Learning Research **23**(47), 1–33 (2022)
20. Hoogeboom, E., Nielsen, D., Jaini, P., Forré, P., Welling, M.: Argmax flows and multinomial diffusion: Learning categorical distributions. In: Thirty-Fifth Conference on Neural Information Processing Systems (2021)
21. Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., Aila, T.: Analyzing and improving the image quality of StyleGAN. In: CVPR (2020)
22. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
23. Kingma, D.P., Salimans, T., Poole, B., Ho, J.: Variational diffusion models. arXiv preprint arXiv:2107.00630 (2021)
24. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013)
25. Kong, X., Jiang, L., Chang, H., Zhang, H., Hao, Y., Gong, H., Essa, I.: Blt: Bidirectional layout transformer for controllable layout generation. arXiv preprint arXiv:2112.05112 (2021)
26. Kong, X., Zhang, Z., Hovy, E.: Incorporating a local translation mechanism into non-autoregressive translation. arXiv preprint arXiv:2011.06132 (2020)
27. Kynkäänniemi, T., Karras, T., Laine, S., Lehtinen, J., Aila, T.: Improved precision and recall metric for assessing generative models. Advances in Neural Information Processing Systems **32** (2019)
28. Nichol, A., Dhariwal, P.: Improved denoising diffusion probabilistic models. arXiv preprint arXiv:2102.09672 (2021)
29. van den Oord, A., Vinyals, O., Kavukcuoglu, K.: Neural discrete representation learning. In: Guyon, I., von Luxburg, U., Bengio, S., Wallach, H.M., Fergus, R., Vishwanathan, S.V.N., Garnett, R. (eds.) NeurIPS (2017)
30. Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., Sutskever, I.: Zero-shot text-to-image generation. In: Meila, M., Zhang, T. (eds.) ICML (2021)
31. Razavi, A., Van den Oord, A., Vinyals, O.: Generating diverse high-fidelity images with vq-vae-2. Advances in neural information processing systems **32** (2019)
32. Rezende, D., Mohamed, S.: Variational inference with normalizing flows. In: International conference on machine learning. pp. 1530–1538 (2015)
33. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training gans. Advances in neural information processing systems **29** (2016)
34. Sauer, A., Schwarz, K., Geiger, A.: Stylegan-xl: Scaling stylegan to large diverse datasets. arXiv preprint arXiv:2202.00273 (2022)
35. Seff, A., Zhou, W., Damani, F., Doyle, A., Adams, R.P.: Discrete object generation with reversible inductive construction. Advances in Neural Information Processing Systems **32** (2019)
36. Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., Ganguli, S.: Deep unsupervised learning using nonequilibrium thermodynamics. In: International Conference on Machine Learning. pp. 2256–2265. PMLR (2015)
37. Song, J., Meng, C., Ermon, S.: Denoising diffusion implicit models. arXiv preprint arXiv:2010.02502 (2020)

38. Vahdat, A., Kautz, J.: Nvae: A deep hierarchical variational autoencoder. Advances in Neural Information Processing Systems (2020)
39. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)
40. Yu, J., Li, X., Koh, J.Y., Zhang, H., Pang, R., Qin, J., Ku, A., Xu, Y., Baldridge, J., Wu, Y.: Vector-quantized image modeling with improved VQGAN. arXiv preprint arXiv:2110.04627 (2021)
41. Zhang, Z., Ma, J., Zhou, C., Men, R., Li, Z., Ding, M., Tang, J., Zhou, J., Yang, H.: M6-ufc: Unifying multi-modal controls for conditional image synthesis. arXiv preprint arXiv:2105.14211 (2021)