IDRF: INVERSE-DISTILLED REWARD FINE-TUNING OF MASKED DISCRETE DIFFUSION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Masked discrete diffusion models offer a flexible alternative to autoregressive generation, but iterative sampling can be costly, and intractable sequence likelihoods complicate reward fine-tuning. We introduce IDRF, a framework for reward fine-tuning of few-step masked discrete diffusion generators. Starting from a standard reverse-KL-regularized reward objective, IDRF replaces the intractable sequence-level KL penalty with an inverse-distillation regularizer. We prove that the population inverse-distillation loss upper-bounds the KL between complete sequences of any length. IDRF optimizes a trajectory-based surrogate of this loss without reference-model rollouts, so the student keeps its own few-step sampler. We view few-step generation as a finite-horizon Markov decision process and optimize reward with a clipped policy-gradient objective over the student's denoising trajectories. We evaluate IDRF across biological, text, and image domains. In each domain, it reaches high reward with up to $32\times$ fewer sampling steps and avoids the reward hacking that reward-only fine-tuning exhibits.



Figure 1: **IDRF across DNA, images, and text.** Reward-only fine-tuning and IDRF sample with the same budget of 8, 4, and 32 steps. Reward-only fine-tuning hacks the reward: DNA lacks accessibility (ATAC-Acc) due to losing naturalness, images blur, and reviews degenerate into repetition that rarely ends with EOS. IDRF avoids these failures.

1 INTRODUCTION

Masked diffusion models (MDMs) generate sequences by unmasking tokens in parallel, with applications to language, biological sequences, and discrete image representations (Sahoo et al., 2024; Shi et al., 2024; Wang et al., 2025a). Reward fine-tuning adapts these models to task objectives, such as DNA activity, image attributes, or review sentiment. Optimizing reward alone, however, can destroy properties the reward does not measure (Figure 1).

KL regularization toward a pretrained reference provides a standard safeguard (Ziegler et al., 2019; Korbak et al., 2022), but an MDM's sequence likelihood marginalizes over all denoising trajectories and is intractable. Existing reward fine-tuning methods use likelihood surrogates and evidence bounds (Zhao et al., 2025; Wang et al., 2026a), or penalize KL between reverse transitions (Wang et al., 2025a). The latter requires reference evaluations along the rollout and ties the regularizer to its

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

sampling path. We seek a tractable sequence-level KL regularizer that lets the student use its own sampler.

We introduce *Inverse-Distilled Reward Fine-tuning* (IDRF), which combines trajectory-based policy gradients with inverse distillation (Li et al., 2026). The regularizer compares the denoising losses of a frozen reference and a fitted auxiliary on student samples. It requires neither the reference's likelihood nor its rollouts. Both the reward and the regularizer reuse the student's own trajectories (Figure 2), so the regularizer shapes the student's few-step sampler directly. Training requires neither pretraining data nor reward gradients. Our contributions are as follows.

1. We extend the inverse-distillation bound of IDLM to sequences of any length: the population loss upper-bounds the sequence-level KL to the endpoint law of an arbitrary reference, for any student sampler (§ 3.1).
2. We regularize reward fine-tuning with a practical inverse-distillation objective on the student's own trajectories, without reference rollouts. As a byproduct, the fine-tuned model is a few-step generator (§ 3.2).
3. Across DNA, images, and text, IDRF reaches high reward with up to $32\times$ fewer sampling steps than the reference while preserving properties lost under reward-only fine-tuning (§ 5).

2 Background

This section reviews the two ingredients of IDRF. § 2.1 introduces masked diffusion and the NELBO used to pretrain the reference, and § 2.2 describes inverse distillation, which IDRF turns into a regularizer for reward fine-tuning.

2.1 Masked Diffusion

Masked diffusion trains a denoiser to recover hidden tokens from their visible context and generates by filling in a fully masked sequence. We follow the MDLM formulation (Sahoo et al., 2024; Shi et al., 2024), which ties this prediction task to a variational bound.

Let $x_0 \in \mathcal{V}^L$ be a clean sequence and $\mathbf{m} \notin \mathcal{V}$ the mask symbol. At noise level $t \in [0, 1]$, each token is retained with probability α_t and masked otherwise, independently across positions. The resulting corruption distribution is

$$q_t(x_t \mid x_0) = \prod_{\ell=1}^{L} \left[\alpha_t \, \delta_{x_0^\ell}(x_t^\ell) + (1 - \alpha_t) \, \delta_{\mathbf{m}}(x_t^\ell)\right], \tag{1}$$

where δ_a is a point mass at a. The schedule α_t is continuously differentiable and strictly decreasing, with $\alpha_0 = 1$ and $\alpha_1 = 0$.

The denoiser $f^\ell(v \mid x_t)$ predicts the missing token at position ℓ from the visible context of x_t and takes no time input. Positions are masked independently of their values, so the Bayes-optimal prediction depends only on the visible tokens of x_t and is independent of t (Ou et al., 2025; Zheng et al., 2025). Accordingly, time conditioning has minimal effect in practice (Sahoo et al., 2024) and implementations omit it (Ou et al., 2025), as do the pretrained models in all our experiments. MDLM trains the denoiser with the continuous-time negative evidence lower bound (NELBO)

$$\mathcal{L}_{\text{NELBO}}(f, p) = \mathbb{E}_{\substack{x_0 \sim p,\, t \sim U(0,1) \\ x_t \sim q_t(\cdot \mid x_0)}} \left[w_t \sum_{\ell : x_t^\ell = \mathbf{m}} -\log f^\ell(x_0^\ell \mid x_t) \right], \qquad w_t = \frac{-\alpha_t'}{1 - \alpha_t}. \tag{2}$$

Here p is the distribution supplying clean sequences, the data distribution p^* during pretraining, so learning to predict missing tokens optimizes a bound on sequence likelihood.

To generate a sample, the model starts from $x_1 = \mathbf{m}^L$. At each reverse step, some masked positions are selected and their values are drawn from the denoiser's predictions. MDLM's substitution-based (SUBS) parameterization keeps revealed tokens fixed and assigns zero probability to predicting $\mathbf{m}$ as

a clean token. For $0 \le s < t \le 1$, this gives the reverse transition

$$
\begin{aligned}
p_f(x_s \mid x_t) &= \prod_{\ell=1}^{L} p_f^{\ell}(x_s^{\ell} \mid x_t), \\
p_f^{\ell}(\cdot \mid x_t) &= \begin{cases} \delta_{x_t^{\ell}}, & x_t^{\ell} \neq \mathbf{m}, \\ \dfrac{1-\alpha_s}{1-\alpha_t}\,\delta_{\mathbf{m}} + \dfrac{\alpha_s - \alpha_t}{1-\alpha_t}\, f^{\ell}(\cdot \mid x_t), & x_t^{\ell} = \mathbf{m}. \end{cases}
\end{aligned} \tag{3}
$$

Positions are updated independently given x_t, but newly revealed tokens condition later predictions, so the generated tokens depend on one another.

2.2 Inverse Distillation

IDLM uses inverse distillation to train a fast discrete generator from a pretrained denoiser (Li et al., 2026). Pretraining fits a denoiser to a given data distribution. Inverse distillation reverses this direction by fixing the reference denoiser f_{ref} and learning a student distribution p_θ on which its predictions remain optimal. In the ideal setting, a student matching p^* would produce samples that the reference can denoise as well as any model fitted specifically to those samples.

A low reference loss alone is not enough to ensure this match. For example, a student that repeats a single sequence may be easy to denoise despite covering very little of the data distribution. Inverse distillation accounts for how predictable the student's samples are by comparing the reference with the best denoiser for those same samples. This gives the objective

$$
\mathcal{L}_{\mathrm{ID}}(\theta) = \mathcal{L}_{\mathrm{NELBO}}(f_{\mathrm{ref}}, p_\theta) - \min_{f} \mathcal{L}_{\mathrm{NELBO}}(f, p_\theta). \tag{4}
$$

Minimizing this gap encourages the student to produce samples on which refitting the denoiser offers little improvement over the reference.

This objective also has a distribution-matching guarantee. With a Bayes-optimal reference for p^* and an exact inner minimum over a sufficiently expressive denoiser family, IDLM shows (Li et al., 2026, Theorem 3.1)

$$
\mathcal{L}_{\mathrm{ID}}(\theta) \ge D_{\mathrm{KL}}(p_\theta \,\|\, p^*) \ge 0, \tag{5}
$$

so a zero gap means that the student matches the data. IDLM proves this bound for a single token and for a joint reverse process over whole sequences, and leaves the sequence objective used in practice, a sum of per-token denoising losses, to future work (Li et al., 2026, App. D.2). Proposition 1 proves the bound for its population loss and an arbitrary reference.

In practice, a trainable auxiliary denoiser f_ψ approximates the inner minimum by learning to reconstruct masked student samples. Training alternates between fitting the auxiliary and updating the student to reduce the difference between reference and auxiliary losses. The reference remains fixed throughout. IDRF uses this denoising gap as the regularizer for reward fine-tuning.

3 Method

We start from the standard KL-regularized reward objective and use inverse distillation as its tractable regularizer. Figure 2 illustrates the resulting training framework.

3.1 KL-regularized reward fine-tuning via inverse distillation

Given a terminal reward R and a frozen reference distribution p_{ref}, the target objective is

$$
\max_{\theta} \mathbb{E}_{x_0 \sim p_\theta}[R(x_0)] - \lambda D_{\mathrm{KL}}(p_\theta \,\|\, p_{\mathrm{ref}}), \qquad \lambda > 0. \tag{6}
$$

The endpoint law marginalizes over all denoising trajectories ending at x_0 (§ 3.2), so its log-density is generally intractable. We instead use the inverse-distillation gap in Eq. (4). The following proposition makes its connection to the KL between full sequences explicit.

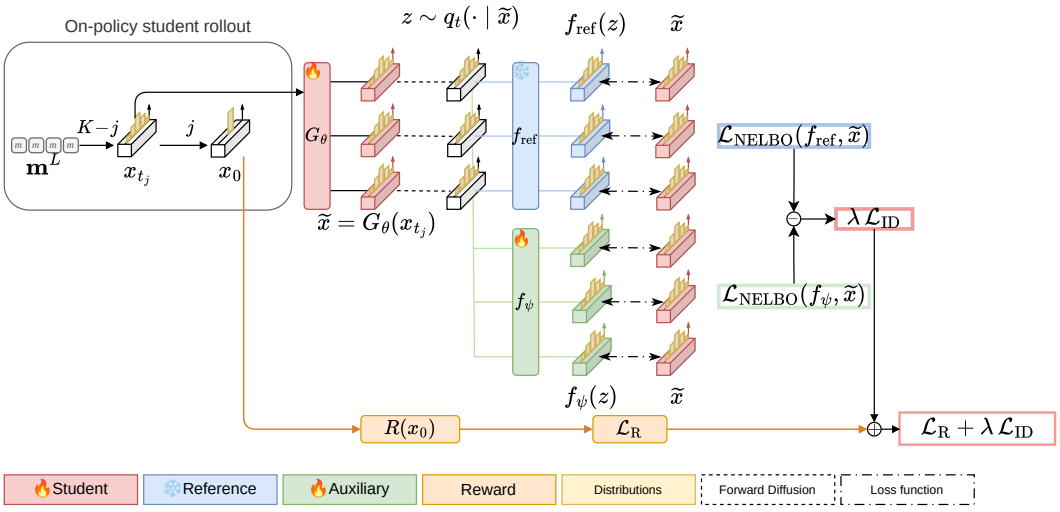


Figure 2: **IDRF training framework.** An on-policy rollout of the student from $\mathbf{m}^L$ provides an intermediate state x_{t_j} and a completed sample x_0. The soft prediction $\widetilde{x} = G_\theta(x_{t_j})$ is freshly corrupted and scored by the frozen reference f_{ref} and the trainable auxiliary f_ψ, whose NELBO gap gives $\mathcal{L}_{\text{ID}}$. The reward of x_0 gives the policy-gradient loss $\mathcal{L}_{\text{R}}$, and the student minimizes $\mathcal{L}_{\text{R}} + \lambda \mathcal{L}_{\text{ID}}$.

Proposition 1 (Sequence-level KL bound). *Let p_θ be any law on $\mathcal{V}^L$. Assume independent forward masking with a continuously differentiable, strictly decreasing schedule with $\alpha_0 = 1$ and $\alpha_1 = 0$. Let p_{ref} be the endpoint law of the continuous-time SUBS reverse process defined by f_{ref}, with unmasking rates $w_t f^\ell_{\text{ref}}(v \mid x_t)$. Assume the reference NELBO is finite and the auxiliary is optimal, that is, its family contains the Bayes denoiser of p_θ on forward-masked states and attains the inner minimum. Then*

$$\mathcal{L}_{\text{ID}}(\theta) := \mathcal{L}_{\text{NELBO}}(f_{\text{ref}}, p_\theta) - \min_\psi \mathcal{L}_{\text{NELBO}}(f_\psi, p_\theta) \;\geq\; D_{\text{KL}}(p_\theta \,\|\, p_{\text{ref}})\,. \tag{7}$$

Equality holds if the reference is the Bayes denoiser of its own endpoint law.

Why the population loss. Proposition 1 holds for any sequence length and any student sampler (Appendix A). Replacing the KL in Eq. (6) by the population loss therefore gives a lower bound on the KL-regularized objective and motivates the regularizer of IDRF.

Practical objective. IDRF optimizes a practical objective built from the population loss. A trained auxiliary takes the place of the Bayes denoiser of p_θ, and the loss is evaluated on the student's predictions $\widetilde{x} = G_\theta(x_{t_j})$ at its rollout states x_{t_j}, $j \sim \mathcal{U}\{1, \ldots, K\}$:

$$\min_\theta \max_\psi \; \mathcal{L}_{\text{R}}(\theta) + \lambda\, \mathbb{E}_{j,\, x_{t_j}} \big[\mathcal{L}_{\text{NELBO}}(f_{\text{ref}}, \widetilde{x}) - \mathcal{L}_{\text{NELBO}}(f_\psi, \widetilde{x})\big], \tag{8}$$

with $\mathcal{L}_{\text{R}}$ and the soft-target NELBO defined in § 3.2. In the experiments, $\mathcal{L}_{\text{ID}}$ and λ_{ID} denote this term and its weight. With an optimal auxiliary, the expected term upper-bounds $D_{\text{KL}}(\bar{q}_\theta \,\|\, p_{\text{ref}})$ for the mixture $\bar{q}_\theta$ of the student's one-step completion laws along its rollouts (Appendix A.1).

3.2 The Few-step Generator as an MDP

We initialize the student denoiser G_θ from the pretrained reference. Like the reference, it maps a partially masked sequence to categorical probabilities over clean tokens at each masked position and takes no time input. Task conditioning is suppressed in the notation. A K-step generator applies G_θ on a fixed grid $1 = t_K > \cdots > t_0 = 0$. At step k, each masked position is revealed with probability $b_k = (\alpha_{t_{k-1}} - \alpha_{t_k})/(1 - \alpha_{t_k})$ and its value is drawn from G_θ, while revealed positions are copied. This is the SUBS transition of Eq. (3) on the grid, and $b_1 = 1$ ends every episode at a

complete sequence x_0 after K network evaluations. Max-confidence decoding instead reveals the most confident masked positions and leaves the rest of the construction unchanged. We formalize the generator as the finite-horizon Markov decision process

$$\begin{cases} \text{state} & s_k = (k,\, x_{t_k}), \qquad s_K = (K,\, \mathbf{m}^L), \\ \text{action} & a_k = x_{t_{k-1}} \in \mathcal{A}(x_{t_k}) = \{y :\ y^\ell = x^\ell_{t_k} \text{ whenever } x^\ell_{t_k} \neq \mathbf{m}\}, \\ \text{policy} & \pi_\theta(a_k \mid s_k) = \displaystyle\prod_{\ell:\, x^\ell_{t_k} = \mathbf{m}} \Big[(1-b_k)\,\delta_{\mathbf{m}}(a^\ell_k) + b_k\, G^\ell_\theta(a^\ell_k \mid x_{t_k})\Big], \\ \text{transition} & s_{k-1} = (k-1,\, a_k), \\ \text{reward} & r(s_k, a_k) = \mathbf{1}[k=1]\, R(a_k). \end{cases} \tag{9}$$

The only reward is received at the final step. A trajectory $\tau = (x_{t_K}, \dots, x_{t_0})$ therefore has probability $P_\theta(\tau) = \prod_{k=1}^K \pi_\theta(x_{t_{k-1}} \mid s_k)$, its endpoint has law $p_\theta(x_0) = \sum_{\tau:\, \tau_0 = x_0} P_\theta(\tau)$, and reward fine-tuning maximizes the expected return $J_R(\theta) = \mathbb{E}_{\tau \sim P_\theta}[R(x_0)]$ of the K-step sampler.

Reward term. Transitions are deterministic and b_k does not depend on θ, so the policy gradient of J_R involves only the tokens revealed at each step, $U_k = \{\ell :\ x^\ell_{t_k} = \mathbf{m} \neq x^\ell_{t_{k-1}}\}$, and never differentiates through discrete draws or through R (Appendix B). Over M rollouts generated by θ_{old}, we optimize the clipped leave-one-out surrogate (Schulman et al., 2017; Kool et al., 2019; Ahmadian et al., 2024; Zhan, 2026)

$$\begin{aligned} \mathcal{L}_{\text{R}}(\theta) &= -\frac{1}{M}\sum_{i=1}^M \sum_{k=1}^K \sum_{\ell \in U_k^{(i)}} \min\big\{\rho_{ik\ell} A_i,\ \text{clip}(\rho_{ik\ell}, 1 \pm \varepsilon)\, A_i\big\}, \\ \rho_{ik\ell} &= \frac{G^\ell_\theta(x_0^{(i),\ell} \mid x^{(i)}_{t_k})}{G^\ell_{\theta_{\text{old}}}(x_0^{(i),\ell} \mid x^{(i)}_{t_k})}, \qquad A_i = R(x_0^{(i)}) - \frac{1}{M-1}\sum_{i' \neq i} R(x_0^{(i')}). \end{aligned} \tag{10}$$

At $\theta = \theta_{\text{old}}$, $-\nabla_\theta \mathcal{L}_{\text{R}}$ is an unbiased estimate of $\nabla_\theta J_R$. For max-confidence decoding, the recorded reveal sets are held fixed.

Inverse distillation on student states. The practical objective in Eq. (8) reuses the rollouts drawn for the reward. The prediction $\widetilde{x}$ serves as a soft target on the simplex (Li et al., 2026), so each cross-entropy in $\mathcal{L}_{\text{NELBO}}$ becomes $-\sum_{v \in \mathcal{V}} \widetilde{x}^\ell(v) \log f^\ell(v \mid z)$ for $f \in \{f_{\text{ref}}, f_\psi\}$ and a fresh corruption z whose visible tokens are drawn from $\widetilde{x}$. At revealed positions, $\widetilde{x}$ is the one-hot copied token. Sampling

Algorithm 1 **IDRF training** — reward fine-tuning of a K-step student

Input: frozen reference f_{ref}; reward R; weight λ; steps K; group size M
$G_\theta \leftarrow f_{\text{ref}},\ f_\psi \leftarrow f_{\text{ref}}$ ▷ zero gap at initialization
for each training iteration **do**
Auxiliary: fit the student's current distribution
  for each auxiliary update **do**
    $x_{t_j} \leftarrow$ partial rollout of the MDP (9), $j \sim \mathcal{U}\{1, \dots, K\}$ ▷ no gradients
    $\psi \leftarrow \psi - \eta\, \nabla_\psi \mathcal{L}_{\text{NELBO}}\big(f_\psi, \text{sg}[G_\theta(x_{t_j})]\big)$ ▷ soft targets
  end for
Student: reward plus inverse distillation
  $\theta_{\text{old}} \leftarrow \theta$; $\tau^{(1)}, \dots, \tau^{(M)} \sim P_{\theta_{\text{old}}}$; record revealed tokens $U_k^{(i)}$ ▷ full K-step rollouts
  $A_i \leftarrow R(x_0^{(i)}) - \frac{1}{M-1}\sum_{i' \neq i} R(x_0^{(i')})$ ▷ Eq. (10)
  for each student update **do**
    $\widetilde{x} \leftarrow G_\theta(x_{t_j})$ with $j \sim \mathcal{U}\{1, \dots, K\}$ on each $\tau^{(i)}$
    $\mathcal{L} \leftarrow \mathcal{L}_{\text{R}}(\theta) + \lambda\big[\mathcal{L}_{\text{NELBO}}(f_{\text{ref}}, \widetilde{x}) - \mathcal{L}_{\text{NELBO}}(f_\psi, \widetilde{x})\big]$ ▷ Eqs. (10), (8)
    $\theta \leftarrow \theta - \eta\, \nabla_\theta \mathcal{L}$
  end for
end for
Output: few-step student G_θ; f_ψ is discarded

j along the whole trajectory distills the student's prediction on exactly the states the sampler visits, with a single auxiliary shared across depths. This provides an effective regularizer and trains the student to sample in few steps. In our ablations, on-policy states work best: forward-masking completed student samples lowers ATAC-Acc from 95.3% to 52.5% in DNA design (Appendix C.4). The regularizer costs one student evaluation plus one pass of the reference and the auxiliary. Gradients flow through the soft targets $\widetilde{x}$, without backpropagation through the sampling trajectory. Algorithm 1 summarizes training.

4 Related Work

Distillation of masked diffusion models. SDTT compresses multiple teacher steps into fewer student steps (Deschenaux & Gulcehre, 2025), while Duo adapts consistency distillation to uniform-state discrete diffusion (Sahoo et al., 2025). Distribution-matching approaches include DiDi-Instruct's integral KL objective (Zheng et al., 2026) and IDLM's inverse distillation (Li et al., 2026).

Reward fine-tuning of masked diffusion models. To handle intractable sequence likelihoods, d1 and wd1 use denoising likelihood approximations (Zhao et al., 2025; Tang et al., 2026), while SPG uses lower and upper evidence bounds (Wang et al., 2026a). Beyond reward fine-tuning, TUBE provides a further upper bound on the evidence of masked diffusion models (Ivanov et al., 2026). Alternatively, AGRPO treats denoising as an MDP with tractable stepwise action probabilities (Zhan, 2026), and TraceRL updates policies along recorded inference trajectories (Wang et al., 2026b). DMPO and Discrete Tilt Matching learn reward-tilted distributions (Zhu et al., 2026c; Chen et al., 2026). DRAKES instead uses differentiable relaxations for biological sequence design when reward gradients are available (Wang et al., 2025a).

Joint distillation and fine-tuning. In continuous diffusion, Diff-Instruct++ regularizes one-step reward optimization with an integral KL (Luo, 2025), DMDR combines distribution matching with reinforcement learning (Jiang et al., 2025), and RTDMD uses reward-tilted distribution matching for few-step flow generators (Huang et al., 2026). For masked image diffusion, Soft-Di[M]O refines a one-step generator through soft embeddings. Its adversarial refinement uses real images, and its reward fine-tuning requires differentiable rewards (Zhu et al., 2026b). IDRF uses inverse distillation as the regularizer of trajectory-based policy gradients, obtains few-step sampling as a byproduct, and requires neither pretraining data nor reward gradients.

5 Experiments

We evaluate IDRF on regulatory-DNA design, CLIP-guided image generation and sentiment steering of review text, and test two claims. First, IDRF attains high reward with a few-step sampler. Second, the inverse-distillation term is what keeps the fine-tuned model close to its reference. Without it, reward-only fine-tuning (RLOO w/o $\mathcal{L}_{\mathrm{ID}}$, $\lambda_{\mathrm{ID}} = 0$) collapses in DNA and text, where likelihood-based metrics miss the failure, and trails IDRF in images.

5.1 Regulatory-DNA design

Setup. Following DRAKES (Wang et al., 2025a), we generate 200-bp regulatory sequences and optimize predicted enhancer activity in HepG2 cells. IDRF students use $\lambda_{\mathrm{ID}} = 0.2$ and sample in $K = 8$ steps with ancestral or max-confidence decoding (Appendix C.2). We report activity under a held-out oracle (**Pred-Activity**), the percentage of sequences predicted accessible in HepG2 (**ATAC-Acc**), 3**-mer correlation** with highly active natural enhancers, **Log-Lik** under the pretrained model, and per-sequence nucleotide **Entropy** (Appendix C.1). We re-evaluate the DRAKES baselines, namely classifier guidance (CG, Nisonoff et al., 2025), SMC and TDS (Wu et al., 2023a), classifier-free guidance (CFG, Ho & Salimans, 2022) and DRAKES with and without its KL term, and the released SEPO checkpoints (Zekri & Boullé, 2025) on 640 sequences for each of three sampling seeds, and Appendix C.3 reconciles our SEPO ATAC-Acc with the published value. The rows for D^2-DPO (Borso et al., 2025), VRPO (Zhu et al., 2026a), DDPP-IS (Rector-Brooks et al., 2025), diffu-GRPO (Zhao et al., 2025) and SDPO, marked *, are reported by Han et al. (2026).

Table 1: **Regulatory-DNA design.** Unmarked values are our measurements, mean (sample SD) over three sampling seeds. † marks the released SEPO checkpoint scored with the HepG2 ATAC output (Appendix C.3), and ∗ results reported by Han et al. (2026), whose released evaluation samples in 128 steps; GF is gradient flow. Bold is best and underline second best among our measurements, excluding the reward-only ablation; ∗ rows and Log-Lik are unranked.

Method	Pred-Activity ↑	ATAC-Acc HepG2 (%) ↑	3-mer Corr. ↑	Log-Lik	Entropy (nats)	Steps↓
Reference	0.18 (0.02)	2.0 (0.4)	−0.042 (0.010)	−262 (0.1)	1.340 (0.001)	128
CG	1.56 (0.02)	0.0 (0.0)	−0.057 (0.002)	−269 (0.2)	1.332 (<0.001)	128
SMC	4.69 (0.48)	49.7 (6.4)	<u>0.880</u> (0.014)	−256 (1.0)	1.299 (0.012)	128
TDS	4.54 (0.55)	33.4 (10.8)	0.822 (0.130)	−257 (1.7)	1.313 (0.024)	128
CFG	5.02 (0.08)	91.7 (0.9)	0.743 (0.004)	−265 (0.7)	1.259 (0.003)	128
DRAKES	5.61 (0.07)	<u>92.3</u> (0.6)	**0.887** (0.001)	−264 (0.3)	1.293 (0.002)	128
DRAKES w/o KL	6.41 (0.03)	82.8 (0.9)	0.307 (0.010)	−281 (0.1)	1.358 (<0.001)	128
SEPO + GF†	**7.66** (0.01)	47.4 (1.8)	0.640 (0.001)	−248 (0.6)	1.177 (0.003)	128
D^2-DPO∗	2.97 (0.03)	35.6 (0.9)	0.944 (0.002)	−252 (0.4)	–	128
VRPO∗	4.60 (0.01)	15.8 (0.2)	0.838 (0.002)	−255 (0.8)	–	128
DDPP-IS∗	4.07 (0.02)	50.0 (0.3)	0.711 (0.001)	−253 (0.9)	–	128
diffu-GRPO∗	5.86 (0.04)	33.0 (0.8)	0.783 (0.001)	−245 (0.4)	–	128
SDPO∗	6.30 (<0.01)	94.8 (<0.1)	0.900 (0.003)	−246 (0.5)	–	128
IDRF (ours)						
ancestral	7.22 (0.01)	**95.3** (0.7)	0.683 (0.001)	−257 (0.1)	1.255 (0.001)	8
max-conf	<u>7.31</u> (0.03)	<u>92.3</u> (0.5)	0.718 (0.003)	−253 (0.3)	1.243 (0.002)	8
RLOO w/o $\mathcal{L}_{\mathrm{ID}}$	7.68 (<0.01)	37.8 (5.5)	0.378 (0.004)	−204 (0.6)	1.036 (<0.001)	8
RLOO w/o $\mathcal{L}_{\mathrm{ID}}$	7.43 (<0.01)	0.1 (0.1)	0.240 (<0.001)	−163 (0.1)	1.072 (<0.001)	128

Accessible, high-activity sequences in eight steps. With eight steps, IDRF attains the top ATAC-Acc in Table 1 with ancestral decoding (95.3%, on par with 94.8% for SDPO), and its max-confidence variant reaches the second-highest activity (7.31). Every baseline samples in 128 steps, 16× more than IDRF. The one method with higher activity, SEPO + GF (7.66), keeps only 47.4% of its sequences accessible, and SDPO reaches activity 6.30.

Without inverse distillation, the oracle is exploited. Removing $\mathcal{L}_{\mathrm{ID}}$ at the same eight steps raises activity to 7.68 but keeps only 37.8% ATAC-Acc (against 95.3%), 3-mer correlation 0.38 (against 0.68) and Entropy 1.04, as the samples collapse into CA/CG-rich sequences almost entirely lacking T (Appendix C.5). With the full 128-step sampler the collapse is complete (0.1% ATAC-Acc). In our runs, Log-Lik ranks both collapsed models above every method (−204 and −163), so the likelihood estimate rewards the failure.

A controllable trade-off. Figure 3 sweeps λ_{ID}. Activity decreases monotonically as the regularizer strengthens, while ATAC-Acc rises from 37.8% to 95.3% and 3-mer correlation from 0.38 to 0.68

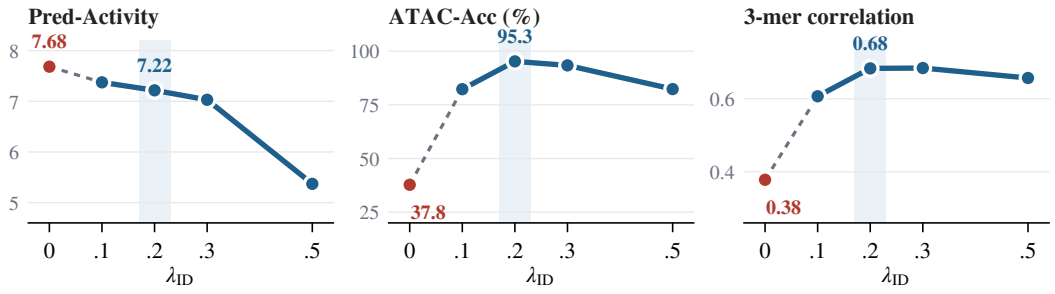


Figure 3: **Regularization sweep of IDRF on regulatory-DNA design** at $K = 8$. Red marks reward-only fine-tuning ($\lambda_{\mathrm{ID}} = 0$), and the shaded band the setting of Table 1.

by $\lambda_{\mathrm{ID}} = 0.2$. A single weight selects the operating point, and larger weights trade activity without further gains.

5.2 CLIP-GUIDED IMAGE GENERATION

We fine-tune a class-conditional MaskGIT (Chang et al., 2022; Besnier et al., 2025) pretrained on ImageNet with CLIP image–text similarity (Radford et al., 2021) as the reward on ten ImageNet classes with attribute prompts (Appendix D), and hold out ImageReward (Xu et al., 2023), PickScore (Kirstain et al., 2023) and HPSv2.1 (Wu et al., 2023b). IDRF and its reward-only ablation both sample in $K = 4$ steps, and each method is evaluated on 1,000 images. At four steps and without guidance, IDRF exceeds the 128-step reference sampled with classifier-free guidance ($w = 3$) on every held-out metric, raising ImageReward from -0.73 to 0.88 (Table 2, Figure 4). Without $\mathcal{L}_{\mathrm{ID}}$, the student reaches lower CLIP (26.12) and ImageReward (-0.49) and falls below the 128-step reference on PickScore and HPSv2.1. ReDGE (Gourevitch et al., 2026) reaches the highest CLIP by optimizing each image at inference, yet stays below IDRF on all held-out metrics.

Table 2: **CLIP-guided MaskGIT generation**, 1,000 images per method (100 per class), mean with 95% confidence interval over images. CLIP is optimized and the remaining metrics are held out. w is the classifier-free guidance weight, and rows without w, including IDRF, use no guidance ($w = 0$). Four-step rows add one cleanup call, and ReDGE adds 400 reward-gradient updates per image.

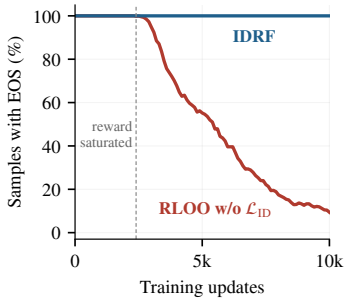
Method	Steps ↓	CLIP ↑	ImageReward ↑	PickScore ↑	HPSv2.1 (×100) ↑
Reference	4	14.00 ± 0.18	-2.17 ± 0.01	18.24 ± 0.03	14.67 ± 0.12
Reference, $w = 3$	128	23.11 ± 0.13	-0.73 ± 0.05	21.10 ± 0.04	25.05 ± 0.12
ReDGE, $w = 3$	20	$\mathbf{37.77} \pm 0.10$	0.30 ± 0.04	21.38 ± 0.03	25.98 ± 0.10
RLOO w/o $\mathcal{L}_{\mathrm{ID}}$	4	26.12 ± 0.10	-0.49 ± 0.03	20.61 ± 0.02	22.57 ± 0.08
IDRF (ours)	4	27.74 ± 0.06	$\mathbf{0.88} \pm 0.02$	$\mathbf{22.11} \pm 0.02$	$\mathbf{27.72} \pm 0.04$



Figure 4: **CLIP-guided MaskGIT samples**, the first sample of each class for each method. Original ImageNet classes appear above the reference row, and reward prompts above the three guided rows. The reference is conditioned on the class alone. ReDGE optimizes the prompt reward at inference, while RLOO and IDRF are fine-tuned on it. IDRF uses four sampling steps plus cleanup.

5.3 SENTIMENT STEERING OF REVIEW TEXT

Text adds a failure mode that fixed-length outputs lack, because the model decides when to stop. We fine-tune an MDLM (Sahoo et al., 2024), trained on Amazon Fashion reviews (Hou et al., 2026) with an explicit EOS token, to maximize $\log P(\text{1 star})$ under a frozen BERT review classifier (Appendix E). We hold out GPT-2-large perplexity (Radford et al., 2019) (**GenPPL**), the mean **length** before the first EOS, the fraction of samples that emit **EOS**, the token entropy of each sample divided by the



RLOO w/o $\mathcal{L}_{\text{ID}}$ — **no EOS, 127 of 127 tokens**

Total junk this this trash is junk!!!!! lies junk not ever even 1 star buying Zero stars and garbage never buy this trash junk deserves 0 stars this junk no one ever buy this junk was junk!! 0 stars NEVER buy this junk 0 [...]

Complete garbage garbage junk this deserve 0 stars this deserves a zero star I never ever I even allowed to give this zero star on this product. 0 stars 0 star this garbage buy this junk ever not ever ever buy this piece of [...]

IDRF (ours) — **32 steps**

This is absolutely junk!!! I purchased this brand and the first time I zipper it up the zipper broke! Please do not purchase this piece of junk! A waste of money!!!! I will never purchase from this company again!!!

This is garbage! It is such horrible quality! I needed this floppy hat for a work event. It got folded in the washing machine and had dark scratches on the brim! Save your money! Don't waste more money on this!

Figure 5: **Reward-only fine-tuning abandons the review format.** The left panel tracks samples that emit EOS at 32 steps. RLOO saturates the reward at update 2,400 (dashed) and IDRF by 2,700, after which RLOO collapses. The right panel shows samples after 10,000 updates, and the RLOO samples are typical of their run, with two thirds of their words repeated.

log of its length (**Norm. entropy**, equal to 1 when no token repeats), and **GM**, the gradient-moment distance of Hoogeboom et al. (2026) to 1,000 held-out real one-star reviews (Appendix E.1).

Reward-only fine-tuning abandons the review format. Both RLOO without $\mathcal{L}_{\text{ID}}$ and IDRF saturate the reward within 2,700 updates (Figure 5). RLOO keeps optimizing past that point by degenerating. Of its 1,000 samples, 906 never emit EOS and run to the 127-token limit as strings of repeated slogans, and normalized entropy drops to 0.72, against 0.95 for real reviews (Table 3). GenPPL does not register the failure and ranks RLOO above the 128-step reference (76.9 versus 91.7).

IDRF preserves the review format in 4–8× fewer steps. IDRF terminates every sample, writes complete reviews (44 tokens on average, against 32 for real one-star reviews) and keeps normalized entropy at 0.90–0.92. At nearly the same reward, its GM to real one-star reviews is 2.6× lower than that of RLOO (10.9 versus 28.8). Neither can approach zero, since real one-star reviews score only −0.95 under the reward, so GM measures how far each method departs from real text to attain it. At 32 steps, IDRF also reaches lower GenPPL than its reference at 128.

Table 3: **Sentiment steering on Amazon Fashion reviews.** $\log P(1\star)$ is optimized and the rest held out. GM (± s.e.) is against held-out real one-star reviews. Bold is the best fine-tuned model.

Method	Steps↓	$\log P(1\star)$ ↑	GenPPL↓	Length	EOS (%)↑	Norm. ent.↑	GM↓
Real one-star	–	−0.950	53.3	32.2	100.0	0.948	0.03 (0.26)
MDLM reference	128	−5.070	91.7	27.8	100.0	0.955	2.45 (0.21)
RLOO w/o $\mathcal{L}_{\text{ID}}$	32	**−0.008**	76.9	126.8	9.4	0.719	28.81 (0.45)
IDRF (ours)	32	−0.044	**65.4**	44.1	**100.0**	0.902	**10.86** (0.50)
IDRF (ours)	16	−0.031	107.3	37.4	**100.0**	**0.922**	11.44 (0.46)

6 Limitations and Future Work

Our guarantee covers the population loss, while the practical objective relies on a trained auxiliary and on the student's own rollout states, so tightening the link between the two remains open. The auxiliary also adds training cost. On MaskGIT, we observed that IDRF samples are less diverse than those of the reference. We hypothesize that inverse distillation into a few-step sampler itself reduces diversity on images, a domain where prior inverse-distillation work has not been evaluated (Li et al., 2026). Testing this hypothesis and improving diversity are left for future work. Scaling IDRF to larger models and other domains is a further direction.

AI USE STATEMENT

AI tools assisted with language editing, code implementation, computational checks, and mathematical analysis during preparation. The authors are responsible for the claims, results, and final manuscript.

ETHICS STATEMENT

This work develops a method for reward fine-tuning of masked diffusion models and involves no human subjects. All experiments use public datasets and pretrained models: the enhancer activity data of Gosai et al. (2024), the ImageNet-trained MaskGIT (Chang et al., 2022), and the Amazon Reviews 2023 corpus (Hou et al., 2026). We do not redistribute review text. The DNA sequences are scored only by predictive models and were not synthesized or tested experimentally. Reward fine-tuning is dual-use. In particular, steering text toward a target sentiment could be misused to produce deceptive reviews at scale, and our sentiment experiment serves only as a benchmark for reward hacking.

REPRODUCIBILITY STATEMENT

The supplementary material contains anonymized code for all three experiments, together with the configurations and evaluation scripts behind the reported results. Appendix A gives complete proofs of the theoretical results. Appendices C, D and E list the models, data processing, hyperparameters, sampling settings, evaluation metrics and seeds, and Appendix C.3 documents the provenance of every baseline number.

REFERENCES

Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Olivier Pietquin, Ahmet Üstün, and Sara Hooker. Back to basics: Revisiting REINFORCE-style optimization for learning from human feedback in LLMs. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2024. doi: 10.18653/v1/2024.acl-long.662. URL https://aclanthology.org/2024.acl-long.662/.

Victor Besnier, Mickael Chen, David Hurych, Eduardo Valle, and Matthieu Cord. Halton scheduler for masked generative image transformer. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=RDVrlWAb7K.

Umberto Borso, Davide Paglieri, Jude Wells, and Tim Rocktäschel. Preference-based alignment of discrete diffusion models. *arXiv preprint arXiv:2503.08295*, 2025. URL https://arxiv.org/abs/2503.08295.

Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T. Freeman. MaskGIT: Masked generative image transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

Yuyuan Chen, Shiyi Wang, Peter Potaptchik, Jaeyeon Kim, and Michael S. Albergo. Discrete tilt matching. In *International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2604.18739.

Justin Deschenaux and Caglar Gulcehre. Beyond autoregression: Fast LLMs via self-distillation through time. In *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2410.21035.

Sager J. Gosai, Rodrigo I. Castro, Natalia Fuentes, John C. Butts, Kousuke Mouri, Michael Alasoadura, Susan Kales, Thanh Thanh L. Nguyen, Ramil R. Noche, Arya S. Rao, Mary T. Joy, Pardis C. Sabeti, Steven K. Reilly, and Ryan Tewhey. Machine-guided design of cell-type-targeting *cis*-regulatory elements. *Nature*, 634:1211–1220, 2024. doi: 10.1038/s41586-024-08070-z.

Samson Gourevitch, Alain Durmus, Eric Moulines, Jimmy Olsson, and Yazid Janati. Categorical reparameterization with denoising diffusion models. *arXiv preprint arXiv:2601.00781*, 2026. URL https://arxiv.org/abs/2601.00781.

Jiaqi Han, Austin Wang, Minkai Xu, Wenda Chu, Meihua Dang, Haotian Ye, Huayu Chen, Yisong Yue, and Stefano Ermon. Discrete diffusion trajectory alignment via stepwise decomposition. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/0922ceebb39f1c76e64ef03d0885fa82-Paper-Conference.pdf.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. URL https://arxiv.org/abs/2207.12598.

Emiel Hoogeboom, David Ruhe, Jonathan Heek, Thomas Mensink, and Tim Salimans. Beyond single tokens: Distilling discrete diffusion models via discrete MMD. *arXiv preprint arXiv:2603.20155*, 2026. URL https://arxiv.org/abs/2603.20155.

Yupeng Hou, Jiacheng Li, Xiangjun Fu, Zhankui He, An Yan, Xiusi Chen, and Julian McAuley. Bridging language and items for retrieval and recommendation: Benchmarking LLMs as semantic encoders. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2026. URL https://aclanthology.org/2026.acl-long.147/.

Yushi Huang, Xiangxin Zhou, Ruoyu Wang, Chi Zhang, Jun Zhang, and Tianyu Pang. Reinforcing few-step generators via reward-tilted distribution matching. *arXiv preprint arXiv:2605.26108*, 2026. URL https://arxiv.org/abs/2605.26108.

Arseny Ivanov, Sergei Kholkin, Vladislav Gromadskii, Grigoriy Ksenofontov, Ivan Oseledets, and Alexander Korotin. TUBE: Tangent upper bound on evidence for discrete diffusion language models. *arXiv preprint arXiv:2605.24292*, 2026. URL https://arxiv.org/abs/2605.24292.

Dengyang Jiang, Dongyang Liu, Zanyi Wang, Qilong Wu, Liuzhuozheng Li, Hengzhuang Li, Xin Jin, David Liu, Changsheng Lu, Zhen Li, Bo Zhang, Mengmeng Wang, Steven Hoi, Peng Gao, and Harry Yang. Distribution matching distillation meets reinforcement learning. *arXiv preprint arXiv:2511.13649*, 2025. URL https://arxiv.org/abs/2511.13649. Accepted at ECCV 2026.

Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-Pic: An open dataset of user preferences for text-to-image generation. In *Advances in Neural Information Processing Systems*, 2023.

Wouter Kool, Herke van Hoof, and Max Welling. Buy 4 REINFORCE samples, get a baseline for free! In *ICLR Workshop on Deep Reinforcement Learning Meets Structured Prediction*, 2019. URL https://openreview.net/forum?id=r1lgTGL5DE.

Tomasz Korbak, Ethan Perez, and Christopher Buckley. RL with KL penalties is better viewed as Bayesian inference. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pp. 1083–1091, 2022. doi: 10.18653/v1/2022.findings-emnlp.77. URL https://aclanthology.org/2022.findings-emnlp.77/.

David Li, Nikita Gushchin, Dmitry Abulkhanov, Eric Moulines, Ivan Oseledets, Maxim Panov, and Alexander Korotin. IDLM: Inverse-distilled diffusion language models. In *International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2602.19066. arXiv:2602.19066v2.

Weijian Luo. Diff-Instruct++: Training one-step text-to-image generator model to align with human preferences. *Transactions on Machine Learning Research*, 2025. URL https://arxiv.org/abs/2410.18881.

Hunter Nisonoff, Junhao Xiong, Stephan Allenspach, and Jennifer Listgarten. Unlocking guidance for discrete state-space diffusion and flow models. In *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2406.01572.

Jingyang Ou, Shen Nie, Kaiwen Xue, Fengqi Zhu, Jiacheng Sun, Zhenguo Li, and Chongxuan Li. Your absorbing discrete diffusion secretly models the conditional distributions of clean data. In *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2406.03736.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI Technical Report*, 2019.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, 2021.

Jarrid Rector-Brooks, Mohsin Hasan, Zhangzhi Peng, Zachary Quinn, Chenghao Liu, Sarthak Mittal, Nouha Dziri, Michael Bronstein, Yoshua Bengio, Pranam Chatterjee, Alexander Tong, and Avishek Joey Bose. Steering masked discrete diffusion models via discrete denoising posterior prediction. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=Ombm8S40zN.

Subham Sekhar Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T. Chiu, Alexander M. Rush, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. In *Advances in Neural Information Processing Systems*, 2024. URL https://arxiv.org/abs/2406.07524.

Subham Sekhar Sahoo, Justin Deschenaux, Aaron Gokaslan, Guanghan Wang, Justin Chiu, and Volodymyr Kuleshov. The diffusion duality. In *International Conference on Machine Learning*, 2025. URL https://arxiv.org/abs/2506.10892.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Jiaxin Shi, Kehang Han, Zhe Wang, Arnaud Doucet, and Michalis K. Titsias. Simplified and generalized masked diffusion for discrete data. In *Advances in Neural Information Processing Systems*, 2024. URL https://arxiv.org/abs/2406.04329.

Xiaohang Tang, Rares Dolga, Sangwoong Yoon, and Ilija Bogunovic. wd1: Weighted policy optimization for reasoning in diffusion language models. In *International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2507.08838.

Chenyu Wang, Masatoshi Uehara, Yichun He, Amy Wang, Tommaso Biancalani, Avantika Lal, Tommi Jaakkola, Sergey Levine, Hanchen Wang, and Aviv Regev. Fine-tuning discrete diffusion models via reward optimization with applications to DNA and protein design. In *International Conference on Learning Representations*, 2025a. URL https://arxiv.org/abs/2410.13643.

Chenyu Wang, Paria Rashidinejad, DiJia Su, Song Jiang, Sid Wang, Siyan Zhao, Cai Zhou, Shannon Zejiang Shen, Feiyu Chen, Tommi Jaakkola, Yuandong Tian, and Bo Liu. SPG: Sandwiched policy gradient for masked diffusion language models. In *International Conference on Learning Representations*, 2026a. URL https://arxiv.org/abs/2510.09541.

Guanghan Wang, Yair Schiff, Subham Sekhar Sahoo, and Volodymyr Kuleshov. Remasking discrete diffusion models with inference-time scaling. In *Advances in Neural Information Processing Systems*, 2025b. URL https://arxiv.org/abs/2503.00307.

Yinjie Wang, Ling Yang, Bowen Li, Ye Tian, Ke Shen, and Mengdi Wang. Revolutionizing reinforcement learning framework for diffusion large language models. In *International Conference on Learning Representations*, 2026b. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/a61023ce36d21010f1423304f8ec49af-Abstract-Conference.html.

Luhuan Wu, Brian L. Trippe, Christian A. Naesseth, David M. Blei, and John P. Cunningham. Practical and asymptotically exact conditional sampling in diffusion models. In *Advances in Neural Information Processing Systems*, 2023a. URL https://arxiv.org/abs/2306.17775.

Xiaoshi Wu, Yiming Hao, Keqiang Sun, Yixiong Chen, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. *arXiv preprint arXiv:2306.09341*, 2023b.

Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. ImageReward: Learning and evaluating human preferences for text-to-image generation. In *Advances in Neural Information Processing Systems*, 2023.

Oussama Zekri and Nicolas Boullé. Fine-tuning discrete diffusion models with policy gradient methods. In *Advances in Neural Information Processing Systems*, 2025. URL https://arxiv.org/abs/2502.01384.

Anthony Zhan. Simple policy gradients for reasoning with diffusion language models. In *International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2510.04019.

Siyan Zhao, Devaansh Gupta, Qinqing Zheng, and Aditya Grover. d1: Scaling reasoning in diffusion large language models via reinforcement learning. In *Advances in Neural Information Processing Systems*, 2025. URL https://arxiv.org/abs/2504.12216.

Haoyang Zheng, Xinyang Liu, Cindy Xiangrui Kong, Nan Jiang, Zheyuan Hu, Weijian Luo, Wei Deng, and Guang Lin. Ultra-fast language generation via discrete diffusion divergence instruct. In *International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2509.25035.

Kaiwen Zheng, Yongxin Chen, Hanzi Mao, Ming-Yu Liu, Jun Zhu, and Qinsheng Zhang. Masked diffusion models are secretly time-agnostic masked models and exploit inaccurate categorical sampling. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2409.02908.

Fengqi Zhu, Rongzhen Wang, Shen Nie, Xiaolu Zhang, Chunwei Wu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. LLaDA 1.5: Variance-reduced preference optimization for large language diffusion models. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 11425–11460, 2026a. doi: 10.18653/v1/2026.acl-long.524. URL https://aclanthology.org/2026.acl-long.524/.

Yuanzhi Zhu, Xi Wang, Stéphane Lathuilière, and Vicky Kalogeiton. Soft-Di[M]O: Improving one-step discrete image generation with soft embeddings. In *International Conference on Learning Representations*, 2026b. URL https://arxiv.org/abs/2509.22925.

Yuchen Zhu, Wei Guo, Jaemoo Choi, Petr Molodyk, Bo Yuan, Molei Tao, and Yongxin Chen. Enhancing reasoning for diffusion LLMs via distribution matching policy optimization. In *International Conference on Machine Learning*, 2026c. URL https://arxiv.org/abs/2510.08233.

Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019. URL https://arxiv.org/abs/1909.08593.

A SEQUENCE-LEVEL INVERSE-DISTILLATION BOUND

We prove Proposition 1 by expressing masked denoising as autoregressive prediction along a random reveal order. The student enters only through its distribution p_θ of completed sequences, in which its generation procedure, including any intermediate states or trajectories, is marginalized. The masking used below is drawn independently after a clean sample is obtained, so it describes the population denoising loss while leaving the student free to use any sampler. As in Proposition 1, the auxiliary is optimal.

Proof. Each masked position in continuous-time SUBS is revealed at total rate w_t, because the denoiser's categorical probabilities sum to one. We realize these events with independent clocks T_ℓ, $\Pr(T_\ell \le t) = 1 - \alpha_t$, whose reverse-time hazard is $(-\alpha_t')/(1-\alpha_t) = w_t$. The clocks are almost surely distinct. Let σ be their order of revelation, which is uniform over permutations by exchangeability (Shi et al., 2024). Because the denoiser takes no time input, the generated values depend on the order of the events and not on their times.

For a clean sequence x, let $z_k(x,\sigma)$ retain the values at positions $\sigma(1),\dots,\sigma(k)$ and mask the rest, so $z_0 = \mathbf{m}^L$ and $z_L = x$. Revealing positions in order σ and drawing each from f defines

$$\pi_{f,\sigma}(x) = \prod_{k=1}^{L} f^{\sigma(k)}\big(x^{\sigma(k)} \mid z_{k-1}(x,\sigma)\big), \qquad p_f(x) = \mathbb{E}_\sigma\, \pi_{f,\sigma}(x), \tag{11}$$

with σ uniform. The second expression is the continuous-time endpoint law, and in particular $p_{f_{\mathrm{ref}}} = p_{\mathrm{ref}}$. Dependence between generated tokens is captured by the growing visible context in each factor.

The NELBO in Eq. (2) is the expected log-loss of these draws. Conditioned on $T_\ell = t$, every other position is independently visible with probability α_t just before position ℓ is revealed, which is exactly the forward-masked context at time t with ℓ masked. The event density $-\alpha_t' = w_t(1-\alpha_t)$ equals the NELBO weight times the probability that ℓ is masked. Summing over positions gives the any-order autoregressive form of the NELBO (Ou et al., 2025),

$$\mathcal{L}_{\mathrm{NELBO}}(f,p) = \mathbb{E}_{x\sim p,\,\sigma}\Big[\sum_{k=1}^{L} -\log f^{\sigma(k)}\big(x^{\sigma(k)} \mid z_{k-1}(x,\sigma)\big)\Big] = \mathbb{E}_{x\sim p,\,\sigma}\big[-\log \pi_{f,\sigma}(x)\big], \tag{12}$$

with x and σ independent. The schedule drops out, since time serves only as an integration variable. The cross-entropy decomposition then yields

$$\mathcal{L}_{\mathrm{NELBO}}(f,p_\theta) = H(p_\theta) + \mathbb{E}_\sigma D_{\mathrm{KL}}(p_\theta \,\|\, \pi_{f,\sigma})\,. \tag{13}$$

Let $V(z)$ be the visible positions of a masked state z. All clean sequences agreeing with z produce this mask pattern with the same probability, which cancels in Bayes' rule, so the optimal auxiliary prediction at a masked position ℓ is

$$f_\theta^{\mathrm{B},\ell}(v \mid z) = \Pr_{x\sim p_\theta}\big(x^\ell = v \mid x^{V(z)} = z^{V(z)}\big). \tag{14}$$

Along any reveal order its predictions are the chain-rule conditionals of p_θ, so $\pi_{f_\theta^{\mathrm{B}},\sigma} = p_\theta$ for every σ. By the assumed expressiveness of the auxiliary family, the minimum in Eq. (13) is therefore $H(p_\theta)$. Subtracting it and applying convexity of KL in its second argument gives

$$\mathcal{L}_{\mathrm{ID}}(\theta) = \mathbb{E}_\sigma D_{\mathrm{KL}}(p_\theta \,\|\, \pi_{f_{\mathrm{ref}},\sigma}) \;\ge\; D_{\mathrm{KL}}(p_\theta \,\|\, \mathbb{E}_\sigma \pi_{f_{\mathrm{ref}},\sigma}) = D_{\mathrm{KL}}(p_\theta \,\|\, p_{\mathrm{ref}})\,. \tag{15}$$

The finite reference NELBO ensures that the gap is finite and $p_\theta \ll p_{\mathrm{ref}}$. If the reference is the Bayes denoiser of p_{ref}, the same chain rule gives $\pi_{f_{\mathrm{ref}},\sigma} = p_{\mathrm{ref}}$ for every σ, and equality holds. □

The argument extends verbatim to time-conditioned denoisers by carrying the event times $T_{\sigma(k)}$ alongside σ through Eqs. (11)–(15). Equation (14) shows that the optimal auxiliary is time-agnostic in either case.

A.1 THE PRACTICAL OBJECTIVE

Fix θ, let $J \sim \mathcal{U}\{1, \dots, K\}$, and let $X \mid J = j$ follow the law $\rho_{\theta,j}$ of the rollout state x_{t_j} of the current student. For a state x with prediction $\widetilde{x} = G_\theta(x)$, one-hot at revealed positions, the one-step completion law and its mixture over depths are

$$Q_{\theta,x}(y) = \prod_{\ell=1}^{L} \widetilde{x}^\ell(y^\ell), \qquad q_j = \mathbb{E}_{X\sim\rho_{\theta,j}} Q_{\theta,X}, \qquad \bar{q}_\theta = \frac{1}{K}\sum_{j=1}^{K} q_j. \tag{16}$$

The soft-target NELBO of Section 3.2 is

$$S(f, \widetilde{x}) = \mathbb{E}_{t,M,Y^V}\Big[w_t \sum_{\ell\in M} -\sum_{v\in\mathcal{V}} \widetilde{x}^\ell(v) \log f^\ell(v \mid Z)\Big], \tag{17}$$

where $t \sim U(0,1)$, each position enters the masked set M independently with probability $1-\alpha_t$, the tokens Y^ℓ at the visible positions $\ell \in V$ are drawn independently from $\widetilde{x}^\ell$, and Z carries Y^V on V and $\mathbf{m}$ on M.

Corollary 1. *Assume the conditions of Proposition 1 with p_θ replaced by $\bar{q}_\theta$, and let the auxiliary objective and the regularizer use the same law of (J, X) and the same corruption. Then, for every auxiliary with finite NELBO,*

$$\mathbb{E}_{J,X}\big[S(f_{\mathrm{ref}}, G_\theta(X)) - S(f_\psi, G_\theta(X))\big] = \mathcal{L}_{\mathrm{NELBO}}(f_{\mathrm{ref}}, \bar{q}_\theta) - \mathcal{L}_{\mathrm{NELBO}}(f_\psi, \bar{q}_\theta). \tag{18}$$

If the auxiliary minimizes its objective, this expectation equals the population loss of $\bar{q}_\theta$ and satisfies

$$\mathcal{L}_{\mathrm{NELBO}}(f_{\mathrm{ref}}, \bar{q}_\theta) - \min_\psi \mathcal{L}_{\mathrm{NELBO}}(f_\psi, \bar{q}_\theta) \;\geq\; D_{\mathrm{KL}}(\bar{q}_\theta \,\|\, p_{\mathrm{ref}})\,. \tag{19}$$

For the sampler of Eq. (9)*, $q_1 = p_\theta$.*

Proof. Fix $J = j$ and $X = x$. Under $Q_{\theta,x}$ the positions are independent, and Z depends on Y only through Y^V. For $\ell \in M$, the token Y^ℓ is therefore independent of (t, M, Y^V), and $\mathbb{E}[-\log f^\ell(Y^\ell \mid Z) \mid t, M, Y^V] = -\sum_{v\in\mathcal{V}} \widetilde{x}^\ell(v) \log f^\ell(v \mid Z)$. Hence $S(f, \widetilde{x}) = \mathcal{L}_{\mathrm{NELBO}}(f, Q_{\theta,x})$. The NELBO is an expectation over the clean sequence and is linear in its law, so averaging over X and J gives $\mathbb{E}_{J,X} S(f, G_\theta(X)) = \mathcal{L}_{\mathrm{NELBO}}(f, \bar{q}_\theta)$. Applying this to f_{ref} and f_ψ proves Eq. (18). By the same identity, the auxiliary objective equals $\mathcal{L}_{\mathrm{NELBO}}(f_\psi, \bar{q}_\theta)$, and its minimum over a family containing the Bayes denoiser of $\bar{q}_\theta$ is attained there. Proposition 1 applied to the law $\bar{q}_\theta$ gives Eq. (19). Finally, $b_1 = 1$ in Eq. (9), so the last transition fills every remaining mask independently from $G_\theta^\ell(\cdot \mid x_{t_1})$ and copies revealed tokens. The endpoint given x_{t_1} therefore has law $Q_{\theta,x_{t_1}}$, and averaging over x_{t_1} gives $q_1 = p_\theta$. □

The corollary characterizes the value of the expected loss for rollout states drawn from the current student. Positions are independent given a rollout state, and the mixture $\bar{q}_\theta$ keeps the dependence between tokens.

B SAMPLING AND POLICY-GRADIENT DETAILS

The ancestral derivation uses a fixed schedule and a final transition to $t_0 = 0$. If a numerical implementation stops at a positive noise floor, any remaining masks are filled by a final categorical draw. Its sampled choices and log-probabilities must also be included in the recorded trajectory and reward loss, and this extra evaluation counts toward the sampling cost. For max-confidence sampling, masked positions are ranked by their maximum predicted probability and revealed according to a fixed step budget. Recomputing their probabilities with the recorded reveal sets held fixed gives the surrogate described in Section 3.2.

To verify the policy gradient, expand the path probability of the MDP in Eq. (9). Because G_θ assigns no mass to $\mathbf{m}$, staying masked and being revealed are disjoint events, and $P_\theta(\tau) = c(\tau)\prod_{k=1}^{K}\prod_{\ell\in U_k} G_\theta^\ell(x_0^\ell \mid x_{t_k})$ with $c(\tau) = \prod_{k=1}^{K} b_k^{|U_k|}(1-b_k)^{m_k-|U_k|}$, where m_k is the number

of masked positions in x_{t_k}. The factor $c(\tau)$ is independent of θ. Differentiating $\sum_\tau P_\theta(\tau)R(x_0)$ gives, for any trajectory-independent baseline b,

$$\nabla_\theta J_R(\theta) = \mathbb{E}_{\tau\sim P_\theta}\Big[(R(x_0) - b)\sum_{k=1}^{K}\sum_{\ell\in U_k}\nabla_\theta \log G_\theta^\ell(x_0^\ell \mid x_{t_k})\Big]. \tag{20}$$

The leave-one-out baseline is independent of trajectory i, conditional on the task, so its expected product with that trajectory's score is zero. At $\theta = \theta_{\text{old}}$, every ratio $\rho_{ik\ell}$ in Eq. (10) equals one and clipping is inactive, so differentiating Eq. (10) yields the negative leave-one-out score-function estimate. After batch reuse, the clipped objective acts as a surrogate for the reward gradient.

C Regulatory-DNA design

C.1 Evaluation

Every unmarked row of Table 1 evaluates one fixed checkpoint on 640 sequences for each of three sampling seeds and reports the mean and sample SD of the three per-seed values. Pred-Activity is the median prediction of the held-out activity oracle. ATAC-Acc is the percentage of sequences whose predicted HepG2 accessibility (output 1 of the ATAC oracle) exceeds 0.5. 3-mer correlation is the Pearson correlation between pooled 3-mer counts of the samples and of the 736 enhancers of Gosai et al. (2024) above the 99.9th percentile of measured HepG2 activity. Log-Lik is the median NELBO-based likelihood estimate of the pretrained model, with 128 evaluation steps and one Monte Carlo sample. Entropy averages the per-sequence nucleotide entropy

$$H_n = -\sum_{b\in\{\text{A,C,G,T}\}} p_{n,b}\ln p_{n,b}, \qquad 0 \le H_n \le \ln 4 \approx 1.386, \tag{21}$$

where $p_{n,b}$ is the frequency of base b in sequence n. The $16\times$ ratio in the main text counts reverse steps, 128 for every baseline and 8 for IDRF.

C.2 Training configuration

All IDRF and reward-only runs start the student and the auxiliary from the pretrained DRAKES model, use its log-linear noise schedule, and optimize the DRAKES fine-tuning oracle, while Pred-Activity uses the held-out evaluation oracle. Each iteration takes 4 auxiliary updates on on-policy states, then draws one batch of rollouts and takes 2 clipped student updates on it (Table 4). We report the final checkpoint. The reward-only run at $K = 128$ accumulates gradients over 8 batches of 32 rollouts and forms the leave-one-out baseline within each batch. Training the ancestral IDRF student takes 10 minutes on one NVIDIA RTX PRO 6000 GPU.

Table 4: Training configuration for regulatory-DNA design.

Setting	IDRF (ours)		RLOO w/o $\mathcal{L}_{\text{ID}}$
Sampler	ancestral	max-confidence	ancestral
Sampling steps K	8	8	8 / 128
λ_{ID}	0.2	0.2	0
Rollouts per iteration	256	256	256 / 256 (8×32)
Leave-one-out group	256	256	256 / 32
Optimizer (student, auxiliary)	AdamW, learning rate 10^{-4}, weight decay 0		
Gradient-norm clipping	1.0		
Clipping ε in Eq. (10)	0.2		
Auxiliary / student updates per iteration	4 / 2		
Iterations	500		
$\mathcal{L}_{\text{ID}}$ state	$x_{t_j}, j \sim \mathcal{U}\{1,\dots,K\}$		
Training seed	0		

C.3 Baseline provenance and evaluation discrepancies

Baselines. Reference is the pretrained model without fine-tuning. Classifier guidance (CG, Nisonoff et al., 2025) and two sequential Monte Carlo methods (Wu et al., 2023a) steer it at inference,

SMC with the pretrained model as proposal and the twisted diffusion sampler (TDS) with CG as proposal. Classifier-free guidance (CFG, Ho & Salimans, 2022) samples a separately trained reward-conditioned model with guidance, as in DRAKES. D^2-DPO (Borso et al., 2025) and VRPO (Zhu et al., 2026a) adapt offline preference optimization to discrete diffusion. DRAKES (Wang et al., 2025a) backpropagates the reward through the generated trajectory with Gumbel-softmax relaxations. diffu-GRPO (Zhao et al., 2025) and SEPO (Zekri & Boullé, 2025) are policy gradient methods, with SEPO + GF denoting its gradient flow (GF) variant. DDPP-IS (Rector-Brooks et al., 2025) matches the reward-tilted posterior with importance sampling, and SDPO (Han et al., 2026) decomposes trajectory alignment into stepwise objectives.

Provenance. Table 1 uses our common-protocol measurements for every unmarked row. Table 5 retains the published ATAC-Acc values for transparency. The two columns have different provenance and are not treated as measurements from the same run.

Table 5: **Published and independently measured ATAC-Acc (%).** Published values follow the cited papers. Our values use HepG2 output 1 and 640 sequences per sampling seed. Parentheses give the reported SD for the papers and sample SD across sampling seeds for our measurements.

Method	Published	Our measurement	Evaluated model
SEPO	99.5 (0.2)	8.70 (1.49)	Released checkpoint
SEPO + GF	99.9 (0.09)	47.40 (1.82)	Released checkpoint

ATAC output selection in SEPO. The SEPO paper defines ATAC-Acc for HepG2 (Zekri & Boullé, 2025), but its released `eval_plots.ipynb` selects output 0 in the ATAC plot and threshold calculations. The shared oracle's checkpoint metadata identifies output 0 as GM12878 and output 1 as HepG2. The original DRAKES evaluator uses output 1, as does the separate `eval.ipynb` in SEPO's repository. We use HepG2 output 1 for every method. The released SEPO and SEPO + GF checkpoints then yield 8.70% and 47.40%, respectively, under our sampling protocol. Table 1 reports only SEPO + GF, which scores higher on Pred-Activity, ATAC-Acc, 3-mer correlation and Log-Lik alike. Plain SEPO scores 7.60 (0.01) Pred-Activity, 8.7 (1.5) ATAC-Acc, 0.461 (0.002) 3-mer correlation, −252 (0.1) Log-Lik and 1.185 (0.003) nats of entropy at 128 steps. This identifies a mismatch in the public plotting/evaluation notebook. It does not establish the execution history of every published result, nor explain differences in Activity, 3-mer correlation, or Log-Lik.

C.4 On-policy states

Table 6 replaces rollout states with forward-masked completed samples, masking the student's sample x_0 at the same depth t_j; the soft target is the student's prediction at the masked state. For ancestral decoding both give the same law of mask patterns and differ in how visible values relate to the rest. On a rollout, the visible tokens were drawn first and the student completes them. After forward masking, the visible tokens come from a finished sample, so the student never continues its own partial work. For max-confidence decoding the mask patterns also differ, because the router reveals confident positions first. Under both samplers, ATAC-Acc and 3-mer correlation drop sharply.

Table 6: State on which $\mathcal{L}_{\text{ID}}$ is evaluated in IDRF (ours), regulatory-DNA design at $K = 8$.

Sampler	$\mathcal{L}_{\text{ID}}$ state	Pred-Activity ↑	ATAC-Acc (%) ↑	3-mer Corr. ↑	Log-Lik
Ancestral	on-policy x_{t_j}	7.22	**95.3**	**0.683**	−257
Ancestral	forward-masked x_0	5.89	52.5	0.273	−261
Max-conf	on-policy x_{t_j}	7.31	**92.3**	**0.718**	−253
Max-conf	forward-masked x_0	6.23	42.8	0.408	−224

C.5 Compositional degeneration under reward-only fine-tuning

Table 7 reports the nucleotide composition of the sequences generated by the ancestral IDRF student and the two reward-only ablations of Table 1. Each method contributes 1,920 sequences of 200 nucleotides (640 per sampling seed), saved in FASTA format, a plain-text format with one header

line and one nucleotide string per sequence. Dinucleotide frequencies count adjacent pairs within each sequence.

Table 7: **Reward-only fine-tuning nearly eliminates thymine.** Nucleotide and dinucleotide composition in percent. The pair column is the combined frequency of CG, GC, CA and AC among all dinucleotides, and T-free is the share of sequences without T.

Method	A	C	G	T	CG+GC +CA+AC	T-free
IDRF, $K = 8$	11.70	30.99	42.64	14.67	44.46	0.00
RLOO w/o $\mathcal{L}_{\text{ID}}$, $K = 8$	19.26	47.38	33.36	<0.01	72.66	99.32
RLOO w/o $\mathcal{L}_{\text{ID}}$, $K = 128$	25.39	43.79	30.82	<0.01	79.61	99.74

Reward-only fine-tuning leaves only 13 T bases at $K = 8$ and 5 at $K = 128$ across 1,920 sequences, against 56,344 for IDRF, so 99.3% and 99.7% of its sequences contain no T. The CA/CG-related pairs rise from 44.5% of all dinucleotides under IDRF to 72.7% and 79.6%. Figure 6 shows the first two sequences of sampling seed 0 for each method, without selection.

IDRF (ancestral) $K = 8$

Sequence 1

```
AGGAACTTCGCTGAGGCGGGCGGGGGGCGGGACGCGCGCGCCTCGCGAGGGCGGCGCAATTGGGGCGTGGGTGCGCGGCG
CTGACGACGTCACTGCAGCGGCGCTGTGGCGCCGGCGGCGCGGGGCGCGCTTCGCGACGTCATGTCACGACGGCGTCGCG
ACGTGCGTCTGACTGGGTTAGTCATCGCCCGCTGCGCGTG
```

Sequence 2

```
GAGGGCTCGGCGCATGGGCGTGTACTTTGCGGGGCGGAGCGTCGCGGCGTGCGCGCACGCGCGCGCGGGGCGCGTCACTT
TCGTGTGCGGGCTGAGGTGGGTCACTGCCTAAGGCGGCGGGGGGGGGGGGCGGAGCTTACGTCGTTGTCGGTTACGTCAC
GCGGCGCCGTCATCTGACGCGCGCCGGCGCCGTCCAGGTG
```

RLOO w/o $\mathcal{L}_{\text{ID}}$ $K = 8$

Sequence 1

```
GGGCGGCCGGCGCCCCGGGCCGCCGGCGGAGACGCGGACGCGCAGACGCAGACGCGCACGCACGCACGCACGCACGCACG
CACGCACGCCCGCACGCACCCACCCACGCACGCGCACGCGCGCACGCACCGACGCACGCACCGACGCCGCAGACGCAGAC
ACACGCAGAGGCAGACGCAGAGGCGCAGCCGCCCCCCCCC
```

Sequence 2

```
GGGCGCGGGCGCGGCCGCCGCCGGGGCGGCGGCAGGCGCACGCGCAGACGCAGACGCAGACGCGCCCGCACGCACGCACG
CACGCACGCACCCACGCAGGGCACGCACGCACCCACGCACGCACGCACGCACGCACGCACGCACACACGCACGCACGCAC
GGACGCAGAGACGACGCAGACGCACACGCGCCCCCCCCCC
```

RLOO w/o $\mathcal{L}_{\text{ID}}$ $K = 128$

Sequence 1

```
GGGGGAAGGGGGAAGGCCCCGGGCCGGGCCCGGGCCGGGGCCGCCGCGAGGCAGACGCACAGACGCAGACGCGCACGCGC
ACACGCGCACGCACGCACGCACACACGCACACGCACACGCGCACGCACGCACGCACGCACGCACGCACGCGCACGCACGC
ACGCACACACGCACACACACACACACACACACACACACAC
```

Sequence 2

```
GGGGGAGGGGGGAAAGCGCCGCCGCGGCGGGCCGGCGGCCGCGACGCCGACGCAGACGCAGAGACGCACACGCACGCACG
CGCACGCACGCACGCACGCACGCACGCACGCACGCACGCACACGCACGCACGCACGCACGCACGCACGCACGCACGCACA
CGCACGCACGCACGCACACACACACACACACACACACACA
```

Figure 6: **Unfiltered DNA outputs**, the first two sequences of sampling seed 0 for each method. Line breaks are for display only.

D CLIP-GUIDED MASKGIT

Model and rewards. The reference is the class-conditional Halton-MaskGIT-L transformer (Besnier et al., 2025), a MaskGIT (Chang et al., 2022) pretrained on ImageNet at 384×384 resolution, which generates a 24×24 grid of VQ tokens from a codebook of 16,384 entries. The student and the auxiliary start from it, and the reference and the VQ decoder stay frozen. The model receives only the ImageNet class, and the attribute prompt of Table 9 enters only through the reward, $100\times$ the CLIPScore of a CLIP ViT-L/14 (Radford et al., 2021). ImageReward, PickScore v1 and HPSv2.1 are never optimized.

Training. Table 8 lists the configuration. Each batch holds 16 groups of 4 rollouts, each group for one class drawn uniformly. The student samples with 4 ancestral steps on a linear grid, followed by one cleanup call that fills any remaining masks. The reward loss uses 4 transitions drawn from each recorded trajectory, rescaled to the full trajectory, and each rollout batch serves a single student update, so every ratio in Eq. (10) equals one. The reward-only ablation uses the same configuration with $\lambda_{\mathrm{ID}} = 0$.

Table 8: Training configuration for CLIP-guided MaskGIT.

Setting	Value
Sampling steps K	4 ancestral steps and one cleanup call
λ_{ID}	0.5
Optimizer (student, auxiliary)	AdamW, learning rate 3×10^{-5}, weight decay 0.01
Rollouts per update	64 (16 groups of 4)
Auxiliary / student updates per iteration	1 / 1
Iterations	80
Guidance weight during training	$w = 0$
$\mathcal{L}_{\mathrm{ID}}$ state	$x_{t_j}, j \sim \mathcal{U}\{1, \ldots, K\}$
Precision	bfloat16 transformer, float32 logits
Training seed	20

Evaluation. Each method generates 100 images for each of the ten classes. Scores are averaged within each class and then across classes, and the 95% confidence interval pools the within-class variances with a Student-t quantile. Classifier-free guidance combines logits as $(1 + w)\,\ell_{\mathrm{cond}} - w\,\ell_{\mathrm{uncond}}$. The 128-step reference uses the Halton sampler (Besnier et al., 2025) at temperature 1 with $w = 3$. ReDGE (Gourevitch et al., 2026) uses the ReMDM sampler (Wang et al., 2025b) with 20 network calls and 20 logit updates per step (400 in total, learning rate 0.5, KL weight 1), the same $100\times$ CLIPScore reward computed per image, and $w = 3$.

Table 9: ImageNet classes and CLIP prompts used in the MaskGIT experiments.

ImageNet ID	Class	CLIP prompt
817	Sports car	a blue sports car
864	Tow truck	a red tow truck
546	Electric guitar	a yellow electric guitar
609	Jeep	a black jeep
560	Football helmet	a yellow football helmet
883	Vase	a black vase
444	Tandem bicycle	a silver tandem bicycle
779	School bus	a yellow school bus
555	Fire engine	a red fire truck
388	Giant panda	a giant panda

E SENTIMENT STEERING OF REVIEW TEXT

Data and SFT. We use the Amazon Fashion split of the Amazon Reviews 2023 corpus (Hou et al., 2026). Reviews are deduplicated on exact stripped text, and a deterministic 1% validation split is held out, with no truncation, packing or sentiment filtering. Each training target is BOS + content + EOS repeated to length 128, and all 127 non-BOS positions are noised and supervised. The reward is the log-probability of the one-star class under the frozen classifier `https://huggingface.co/LiYuan/amazon-review-sentiment-analysis`. Starting from the MDLM OpenWebText checkpoint (Sahoo et al., 2024) with the GPT-2 tokenizer, we run EOS-aware SFT for 50,000 steps at learning rate 3×10^{-4} and take the EMA weights. This single checkpoint initializes the frozen reference, the student and the auxiliary denoiser for every reward fine-tuning run reported in Table 3.

Reward fine-tuning. Rollouts use ancestral sampling followed by a stochastic categorical terminal step, so K reverse updates cost $K + 1$ denoiser evaluations. The score-function estimator of Eq. (20) is applied to the full trajectory without subsampling reverse steps. Each rollout batch receives a single

on-policy update, so $\theta = \theta_{\text{old}}$ and the clipping in Eq. (10) is inactive here. Groups contain 8 rollouts per device, for a global batch of 64 across 8 devices. The student and the auxiliary denoiser both use learning rate 10^{-6} with 100 warmup steps, and the auxiliary is updated once per student update. Evaluation runs every 100 updates on $1{,}000$ fresh samples, and weights are checkpointed every 500 updates. The reward and GenPPL are read from the text up to the first EOS, while the RLOO and inverse-distillation losses are computed on the full length-128 buffer, including the EOS tail. Every run starts from the same SFT checkpoint with training seed 0, and IDRF uses $\lambda_{\text{ID}} = 0.01$.

Evaluation. All generated rows of Table 3, including the reference, use the same sampler, with ancestral updates followed by a stochastic categorical terminal step, evaluation seed 1 and $1{,}000$ samples. The real-review rows score $1{,}000$ held-out reviews that are disjoint from the GM reference. **Norm. entropy** is the Shannon entropy of the GPT-2 token frequencies within a sample divided by the log of its length, averaged over samples. Length does not explain the ordering it gives. On prefixes of exactly 20, 32 and 44 tokens, RLOO scores 0.88, 0.85 and 0.82, IDRF at 32 steps 0.94, 0.92 and 0.90, and real one-star reviews 0.96, 0.94 and 0.93. Word-level repetition agrees, with repeated words making up 66% of a median RLOO sample, 14% for IDRF and 9% for real one-star reviews.

E.1 Gradient-moment diagnostic

GM adapts the centered estimator of Hoogeboom et al. (2026, Eq. 14) to variable-length text. For a candidate set and a reference set of $1{,}000$ reviews each, we compute the mean gradient of GPT-2-large's token NLL on two disjoint halves of 50 reviews from each set and take the inner product of the two candidate-minus-reference differences. The reported value averages 10 such estimates, with one standard error across them. Each review contributes its mean token NLL, conditioned on BOS and including one EOS. Gradients are computed in float32 and inner products in float64. The estimator is unbiased for the squared distance between mean gradients, so two samples of the same distribution score zero in expectation.

Table 10 uses two references from the held-out validation split. One holds $1{,}000$ of the $2{,}748$ eligible real one-star reviews, and the other $1{,}000$ of the $21{,}448$ real reviews of every rating. Each reference is compared against a disjoint sample of the same distribution, and both real-versus-real estimates (0.03 and 0.06) are consistent with zero. Against both references, IDRF has a 2.4–$2.6\times$ lower GM than reward-only fine-tuning. One-star and average real reviews differ by only 1.99, far less than the gap between the two fine-tuned models, so this gap reflects the drift that reward-only optimization introduces and inverse distillation reduces. Neither fine-tuned model can score near zero, because real one-star reviews reach only $\log P(1\star) = -0.95$, against -0.04 for IDRF.

Table 10: Gradient moment against two held-out references (lower is closer), $\pm$ one standard error over ten pair estimates. Generated sets are those of Table 3.

Candidate set	Steps ↓	GM vs. real one-star ↓	GM vs. real, all ratings ↓
Real one-star reviews	–	0.03 ± 0.26	1.99 ± 0.29
Real reviews, all ratings	–	–	0.06 ± 0.24
MDLM reference	128	2.45 ± 0.21	0.54 ± 0.21
RLOO w/o $\mathcal{L}_{\text{ID}}$	32	28.81 ± 0.45	29.53 ± 0.47
IDRF (ours)	32	10.86 ± 0.50	12.13 ± 0.56
IDRF (ours)	16	11.44 ± 0.46	12.85 ± 0.37