Locality-Controlled Out-of-Distribution Guidance for Robust Regulatory DNA Sequence Design

Dabin Jeong
Wellcome Sanger Institute
dj16@sanger.ac.uk

Marwane Bourdim
Wellcome Sanger Institute
mb57@sanger.ac.uk

Mo Lotfollahi
Wellcome Sanger Institute
ml10@sanger.ac.uk

Abstract

Designing regulatory DNA sequences for targeted gene-expression control is an important challenge. Recent approaches frame this task as property-conditioned generation, where diffusion models trained on natural sequences are guided toward high predicted activity through reward-based fine-tuning or sampling-time guidance. We show that single-reward evaluation can obscure reward-hacking behavior. By decomposing evaluation into property, novelty, and plausibility, we identify an empirically underexplored region of the evaluation space in which all three objectives are satisfied, but no existing baseline succeeds. We postulate that this gap arises because prior methods treat guidance primarily as a matter of magnitude. In regulatory DNA, however, sequence grammar is spatially nonuniform: motif positions are sensitive to perturbation, while background regions allow more exploration. We therefore propose LOCOGEN by reformulating guidance as a problem of **locality**. LOCOGEN learns an invariant motif mask through sequence-level invariance learning to protect motif positions, while applying OOD damping only to background positions. It operates at sampling time on a frozen diffusion backbone, without retraining the backbone. Across HepG2, K562, and SK-N-SH, LOCOGEN reaches this previously underexplored region of the property–novelty–plausibility space. Code is available at `https://github.com/Lotfollahi-lab/LocoGen`.

1 Introduction

Diffusion models are powerful generative models used in many domains, including images, language, molecules, and biological sequences. Beyond unconditional generation, recent work increasingly focuses on property-conditioned generation, such as generating regulatory DNA sequences that optimize a target property (e.g. gene expression). These advances make diffusion models useful for *de novo* regulatory DNA design, where candidate sequences can be generated and tested in a lab-in-the-loop setting.

We argue that the generated candidates must satisfy three axes simultaneously: (i) high target **property**, (ii) **novelty**, and (iii) **plausibility** —preservation of the natural grammar of regulatory DNA (e.g. motif usage diversity/density). No single axis suffices. High property without plausibility fails to reproduce in the wet lab; high novelty without plausibility drifts off the functional manifold; high property without novelty merely reproduces the training distribution. Only when all three axes are jointly satisfied is a candidate worth its wet-lab budget.

To date, DNA sequence generative models have largely treated candidate generation as a single-objective problem: maximizing the target property. However, when evaluation is decomposed into property, novelty, and plausibility, methods that appear strong under single-axis evaluation reveal trade-offs on the remaining axes (Fig. 1B). Moreover, their apparent property gains often arise not from targeted regulatory improvement, but from distortions of regulatory motif structure, exposing a common limitation of existing guidance methods (§5.3).

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

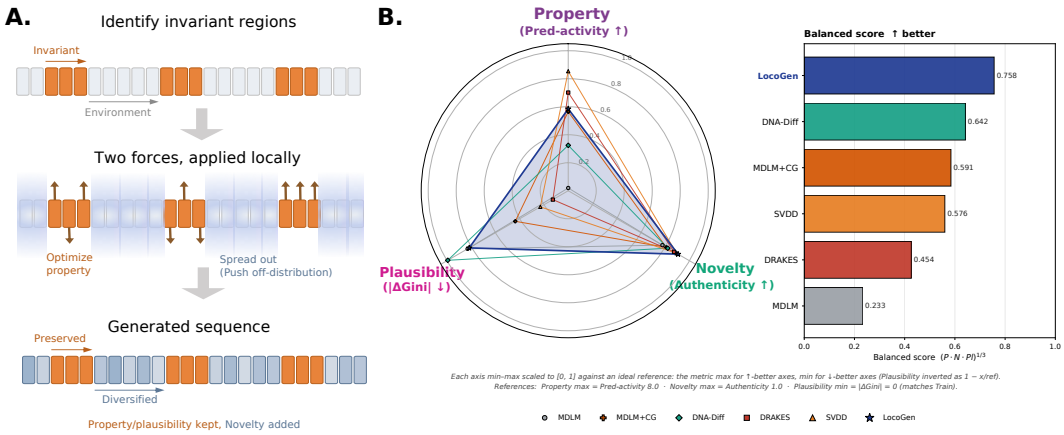

Figure 1: LOCOGEN balances property, novelty, and plausibility.

Although these methods differ in their reward-hacking mechanisms, they all control optimization through a single *scalar magnitude knob* (e.g. guidance scale) without specifying *where* in the sequence the reward should act. This is problematic because regulatory DNA grammar is positional: motif sites are highly constrained, whereas intervening background regions are more permissive. Thus, in the property–novelty–plausibility design space, current baselines fall short of the desired regime: high activity on both readouts, high novelty, and low compositional deviation. We call this unoccupied region the **empty corner**, and ask whether guidance can shift the empirical trade-off landscape toward it.

We propose LOCOGEN, a locality-controlled guidance method for *de novo* regulatory DNA generation (Fig. 1A). Rather than increasing guidance globally, LOCOGEN controls *where* it acts. A masked separator model identifies foreground and background positions with an invariant mask; during sampling, out-of-distribution (OOD) guidance is applied only to background positions. This generates novel sequences in permissive regions while preserving regulatory motifs, thereby improving novelty without sacrificing enhancer-like composition or functional activity.

Contributions.

- **Diagnostic framework.** We introduce a three-axis evaluation framework for regulatory DNA generation: property, novelty, and compositional plausibility. It reveals two reward-hacking modes concealed by single-axis evaluation: predictor-specific motif concentration and global compositional drift in previous DNA sequence generative models.
- **Locality-controlled guidance.** We reformulate sampling-time guidance as a locality problem and instantiate it as localized OOD control with an invariant subsequence mask. The method operates on a frozen backbone, requires no retraining, and exposes per-position spatial control absent from standard classifier-guidance methods.

2 Related Work

Discrete diffusion and flow matching on biological sequences. Two frameworks have emerged for generative modeling of categorical sequences. *Discrete diffusion* operates directly on categorical sequences and learns to denoise progressively masked tokens [Austin et al., 2021, Sahoo et al., 2024, Lou et al., 2024]. *Continuous diffusion and flow matching* instead work in a continuous relaxation of the categorical space, typically the probability simplex whose vertices correspond to the discrete tokens [Stark et al., 2024, Davis et al., 2024] Both have been applied to regulatory DNA design [Stark et al., 2024, DaSilva et al., 2026, Tang et al., 2025]. We build on the discrete diffusion line, using a pretrained masked discrete diffusion language model (MDLM) [Sahoo et al., 2024] as a backbone and steering generation with inference-time guidance.

Reward-based fine-tuning of diffusion backbones. A first family of methods directly fine-tunes the backbone against a reward signal, often framed as reinforcement learning. DRAKES [Wang et al., 2025] backpropagates a differentiable reward to pre-trained masked discrete diffusion model; DRaFT [Clark et al., 2024] applies the same differentiable-reward principle in the continuous setting. When the reward is non-differentiable, policy-gradient or gradient-free alternatives are used instead [Cao et al., 2025]. These methods can each optimize toward a particular property, but they do not expose a post-hoc control knob: changing the target reward generally requires re-specifying the training objective and running a separate fine-tuning pass, even when recent variants reduce its cost or handle multiple objectives jointly. Although recent methods reduce the cost of retraining [Cao et al., 2025], they inherently require backbone retraining for each new objective.

Training-free guidance. Classifier guidance instead allows the guidance signal to be adjusted entirely at inference time through an external scoring network [Uehara et al., 2025, Rojas et al., 2026]. DNA-Diffusion [DaSilva et al., 2026], by contrast, conditions on cell-type labels rather than optimizing a reward, and therefore serves as a non-optimizing reference rather than a competing optimizer. Another method that bypasses backbone retraining is by sampling many candidates and selecting the highest-scoring ones. SVDD [Li et al., 2024] performs soft value-based decoding, evaluating generated candidates at each step and resampling proportionally to a learned value.

3 Preliminary

3.1 Notation

We consider DNA sequences of fixed length L over the nucleotide alphabet $\mathcal{A} = \{A, T, G, C\}$. A DNA sequence is denoted by $X = (x_1, \ldots, x_L) \in \mathcal{A}^L$, where $x_\ell \in \mathcal{A}$ represents the nucleotide at position ℓ. Its one-hot representation is denoted by $\mathbf{x} \in \{0,1\}^{L\times 4}$, where each row $\mathbf{x}_\ell \in \{0,1\}^4$ satisfies $\sum_{k=1}^4 \mathbf{x}_{\ell,k} = 1$.

3.2 Problem formulation

We consider paired observations (X, Y), where $X \in \mathcal{X} = \mathcal{A}^L$ denotes a DNA sequence and $Y \in \mathcal{Y} \subseteq \mathbb{R}$ denotes an associated scalar property (e.g., target gene activity). Let $p_\theta(X)$ denote a base generative model over DNA sequences, and let $p_\phi(Y \mid X)$ denote an auxiliary property predictor. For a target property value $y^\star$, classifier-based guidance aims to reweight the base generative distribution according to the predicted property likelihood as follows,

$$p_{\text{guide}}(X \mid y^\star) \propto p_\theta(X)\, p_\phi(y^\star \mid X)^\lambda, \tag{1}$$

where $\lambda \geq 0$ controls the guidance strength. Since our goal is to generate novel regulatory sequences, effective generation guidance must balance three objectives: (i) achieving the target property $y^\star$, (ii) producing novel sequences that depart from the training distribution rather than memorizing it, and (iii) maintaining plausibility.

3.3 Masked Discrete Diffusion Language Models

MDLM [Sahoo et al., 2024] generates sequences by progressively denoising a fully masked input rather than autoregressively generating tokens. We extend the alphabet with an absorbing mask token [MASK], defining $\mathcal{A}_{\text{mask}} = \mathcal{A} \cup \{[\texttt{MASK}]\}$. Its one-hot representation is denoted by $\mathbf{m} \in \{0,1\}^5$. The masked sequence at diffusion time t is denoted by $\mathbf{z}_t \in \{0,1\}^{L\times 5}$.

During training, MDLM learns to recover clean tokens from a masked sequence $\mathbf{z}_t \in \{0,1\}^{L\times 5}$. In the forward process, each clean token in $\mathbf{x}$ is independently replaced with $\mathbf{m}$ with probability $1 - \alpha_t$, yielding $\mathbf{z}_t$, where α_t decreases monotonically from $\alpha_0 \approx 1$ (clean) to $\alpha_1 \approx 0$ (fully masked). In the reverse process, a denoising network $\mathbf{x}_\theta(\mathbf{z}_t, t) : \{0,1\}^{L\times 5} \times [0,1] \to [0,1]^{L\times 5}$ approximates the clean sequence $\mathbf{x}$ by predicting the categorical distribution over clean nucleotides at each position, assigning zero probability to the mask token.

At inference time, generation proceeds by sampling a less-masked state $\mathbf{z}_s$ from the current state $\mathbf{z}_t$, with $s < t$:

$$p_\theta(\mathbf{z}_{s,\ell} \mid \mathbf{z}_t) = \begin{cases} \delta_{\mathbf{z}_{t,\ell}}(\mathbf{z}_{s,\ell}), & \mathbf{z}_{t,\ell} \neq \mathbf{m}, \\ \text{Categorical}\left(\mathbf{z}_{s,\ell};\ \dfrac{(1-\alpha_s)\,\mathbf{m} + (\alpha_s - \alpha_t)\,\mathbf{x}_{\theta,\ell}(\mathbf{z}_t, t)}{1-\alpha_t}\right), & \mathbf{z}_{t,\ell} = \mathbf{m}. \end{cases} \tag{2}$$

Here, $p_\theta(\mathbf{z}_{s,\ell} \mid \mathbf{z}_t)$ denotes the conditional distribution of the token at position ℓ in the next denoising step while $\delta_{\mathbf{z}_{t,\ell}}(\mathbf{z}_{s,\ell})$ denotes a point mass at the current token, ensuring that previously unmasked positions remain unchanged (See §S1.2.2 for details).

The time $s < t$ corresponds to the next, less-masked diffusion state in the reverse process. Masked positions are either kept as $\mathbf{m}$ or filled in using the model's prediction. Generation terminates when all positions are unmasked. In this work, we use a pretrained unconditional MDLM over fixed-length DNA sequences as the base model and guide its generation toward specific objectives.

3.4 Classifier guidance with OOD control

MOOD [Lee et al., 2023] is a score-based diffusion framework that combines OOD control with property guidance to generate novel molecules with desirable properties beyond the training distribution. It operates on continuous representations of molecular graphs. Let $\mathbf{u}_t \in \mathcal{G}$ denote a noise-perturbed molecular graph representation at diffusion time t, where $\mathcal{G}$ is the continuous space of molecular graph representations, and let $f_\phi : \mathcal{G} \to \mathbb{R}$ denote a differentiable property scorer trained on molecular data. In the score-based diffusion framework, classifier guidance modifies the reverse-time SDE drift by adding the gradient of $p_t(y_p \mid \mathbf{u}_t) \propto \exp(\beta_t f_\phi(\mathbf{u}_t))$,

$$d\mathbf{u}_t = \Big[f_t(\mathbf{u}_t) - g_t^2\Big(\nabla_{\mathbf{u}_t} \log p_t(\mathbf{u}_t) + \beta_t \nabla_{\mathbf{u}_t} f_\phi(\mathbf{u}_t)\Big)\Big]dt + g_t\, d\bar{w}_t, \tag{3}$$

where p_t denotes the marginal density of the forward diffusion process, f_t and g_t are its drift and diffusion coefficients, respectively, $\bar{w}_t$ is a reverse-time Wiener process, and $\beta_t \geq 0$ controls the guidance strength. The SDE is integrated backward in time.

MOOD additionally introduces an OOD-control coefficient $\gamma \in [0, 1)$ that rescales the generative score:

$$d\mathbf{u}_t = \Big[f_t(\mathbf{u}_t) - (1 - \sqrt{\gamma})\, g_t^2 \nabla_{\mathbf{u}_t} \log p_t(\mathbf{u}_t) - \beta_t g_t^2 \nabla_{\mathbf{u}_t} f_\phi(\mathbf{u}_t)\Big]dt + g_t\, d\bar{w}_t. \tag{4}$$

where $\gamma = 0$ generates samples from property-guided diffusion without OOD control, and increasing γ encourages exploration of lower-density regions, thus generating more OOD samples. MOOD applies a single global OOD-control coefficient γ to the generative score across all graph components, without adapting this coefficient to their individual contributions to the target property.

4 Method

We propose LOCOGEN, a localized sequence generation framework that controls both the magnitude and locality of generated sequences. The uniform application of guidance and OOD control in Eq. 4 encodes an implicit assumption: that all sequence positions should be perturbed by the same amount. We argue this assumption is not perfectly suited for regulatory DNA. Functional motifs occupy a small fraction of each sequence and are highly constrained by sequence grammar, while the remaining background is comparatively unconstrained. Pushing motifs off-manifold destroys the very signal that makes a sequence functional, while pushing background off-manifold contributes novelty without sacrificing function.

4.1 Module 1: Masked Separator for Invariant–Environment Decomposition

Problem formulation. We aim to identify subsequences whose contribution to activity is *invariant* across data subpopulations, i.e., positions whose predictive role does not change from one subpopulation to another. A straightforward approach is to train an activity predictor and use saliency, but such scores reflect what a single trained model happens to rely on, without explicitly accounting for invariance. We instead build this requirement into the learning objective: the selected positions are encouraged to predict activity *equally well across subpopulations*.

Approach. Inspired by the invariant-environment decomposition of chemical graphs [Li et al., 2022], we introduce a learned soft mask $M \in [0,1]^L$ that splits each sequence into an invariant part x^{inv} and an environment part x^{env}, with M_ℓ close to 1 marking position ℓ as invariant and close to 0 as environment, jointly satisfying: (a) x^{inv} alone is sufficient to predict activity; (b) its prediction error remains consistent across auto-discovered environments; and (c) M is sparse and contiguous.

Invariant learning for subsequence disentanglement. We jointly train a separator S_ϕ producing the soft mask $M \in [0,1]^L$ and a predictor h_ψ on the invariant subsequence with disjoint weights. The environment encoder g_ω on the environment subsequence is randomly initialized and kept frozen. Keeping g_ω frozen prevents the environment encoder itself from adapting to trivially collapse the environment representations, leaving the separator to determine the partition of sequence information. The environment is still not static since x^{env} depends on the current mask. To prevent information leakage through masked positions, we fill them with a learnable token $b \in \mathbb{R}^K$, where $K = 4$ is the nucleotide alphabet size (initialized uniformly to $1/K$):

$$x^{\text{inv}} = x \odot M_K + \bar{b} \odot (\mathbf{1} - M_K), \quad x^{\text{env}} = x \odot (\mathbf{1} - M_K) + \bar{b} \odot M_K, \tag{5}$$

where $M_K = M \otimes \mathbf{1}_K$ and $\bar{b} = \mathbf{1}_L \otimes b$. The predictor and the environment encoder read out $\hat{y}^{\text{inv}} = h_\psi(x^{\text{inv}})$ and h_ψ's counterpart $g_\omega(x^{\text{env}})$, respectively.

Since environment labels are unavailable, we infer them via k-means clustering with $E{=}3$ clusters on $g_\omega(x^{\text{env}})$, following GIL [Li et al., 2022]. At the start of each epoch, we freeze a snapshot of g_ω, encode a fixed reference batch, and refit the centroids, which are then cached and used to assign environment labels throughout that epoch. Deriving both the centroids and the embeddings from the same snapshot keeps the label assignment self-consistent. See §S1.3.3 for details.

Training objective.

$$\mathcal{L} = \underbrace{\mathcal{L}_{\text{sta}}}_{\text{Sufficiency}} + \lambda_{\text{inv}} \underbrace{\mathcal{L}_{\text{inv}}}_{\text{Invariance/necessity}} + \lambda_{\text{reg}}\, \mathcal{L}_{\text{reg}} + \lambda_{\text{smooth}}\, \mathcal{L}_{\text{smooth}}. \tag{6}$$

Our objective consists of four losses $\mathcal{L}_{\text{sta}}$ for sufficiency, $\mathcal{L}_{\text{inv}}$ for invariant learning via V-REx, $\mathcal{L}_{\text{reg}}$ for density regularization, and $\mathcal{L}_{\text{smooth}}$ for total variation smoothing. $\mathcal{L}_{\text{sta}}$ encourages the predictor to recover y from invariant sequence alone, x^{inv}: $\mathcal{L}_{\text{sta}} = \frac{1}{B}\sum_n (h_\psi(x^{\text{inv}(n)}) - y^{(n)})^2$, where n indexes the B samples of a minibatch. For robust prediction, we adopt variance risk extrapolation (V-REx) [Krueger et al., 2021] to encourage consistent predictive performance across inferred environments. Let $\epsilon^{(n)} = (\hat{y}^{\text{inv}(n)} - y^{(n)})^2$ and $\hat{e}^{(n)} \in \{1, \ldots, E\}$ be the k-means environment label of sample n. With per-environment risk $R_e = \mathbb{E}_{n:\hat{e}^{(n)}=e}[\epsilon^{(n)}]$,

$$\mathcal{L}_{\text{inv}} = \text{Var}_e[R_e]. \tag{7}$$

Together with $\mathcal{L}_{\text{sta}}$, this encourages the predictor to perform *equally well in every environment*: any environment-specific information leaking into x^{inv} would inflate the variance and incur a penalty. $\mathcal{L}_{\text{reg}}$ keeps the average mask density near a target ρ, and $\mathcal{L}_{\text{smooth}}$ is a fused-lasso total variation penalty [Javanmardi et al., 2016] encouraging contiguous selected blocks. See §S1.3.2 for details on loss definition. The trained separator and final centroids are saved and remain frozen during sampling.

4.2 Module 2: Sampler with localized OOD control

Problem. The MOOD framework (§3.4) promotes novel, high-scoring samples by combining sample-level OOD drift with property guidance. For regulatory DNA, this global control is too coarse: without distinguishing constrained motifs from permissive background regions, OOD drift can erase regulatory syntax and compromise activity.

Approach. We let damping act only on positions the separator marks as environment. By restricting OOD control to environment positions, we aim to preserve activity-relevant sequence patterns while encouraging exploration in less constrained regions. This is the localization in LOCOGEN: a structural prior on *where* OOD pressure may act, instead of one global knob trading off the two objectives.

Reverse unmasking process. At each step $t \to s$, the reverse process over nucleotide k at position ℓ is

$$\tilde{p}_\theta(x_s \mid x_t)_{\ell,k} \;\propto\; \underbrace{\exp\big(d_\ell(\gamma) \cdot \log p_\theta\big)}_{\text{(a) localized OOD damping}} \cdot \underbrace{\Delta_{t\to s}}_{\text{(b) MDLM unmask schedule}} \cdot \underbrace{\exp(\eta_t \, \nabla r)}_{\text{(c) property pull}}, \tag{8}$$

where $\Delta_{t\to s} := (\alpha_s - \alpha_t)/(1-\alpha_t)$ is the probability mass that MDLM assigns to the nucleotide channels at a masked position (Eq. 2), and $\nabla r := \nabla_{\hat{x}_0} r_\xi(\hat{x}_0)$ is the gradient of the activity guidance model r_ξ evaluated on the current x_0-prediction $\hat{x}_0$ obtained by softmaxing the nucleotide channels of $\log p_\theta$, following the masked-diffusion parameterization of MDLM. The three factors push three forces in parallel: (a) flattens background nucleotide distributions toward uniform, (b) preserves MDLM's prescribed unmasking rate, and (c) pulls toward high activity

Where OOD control acts. OOD control reduces $\log p_\theta$ by a factor $d_\ell(\gamma) < 1$, flattening the per-position categorical distribution toward uniform and thus injecting randomness. We apply OOD damping to positions identified as environment by the separator, using the predicted mask $M_t = S_\phi(\hat{x}_0)$:

$$d_{\ell,t}(\gamma) = \begin{cases} 1 - \sqrt{\gamma}, & M_{t,\ell} < \tau_M \text{ and } t < \tau_{\text{time}}, \\ 1, & \text{otherwise.} \end{cases} \tag{9}$$

Restricting damping to masked positions keeps already-decoded nucleotides untouched, consistent with the SUBS parameterisation of MDLM, and is activated only in the latter half of sampling, when $\hat{x}_0$ provides a more reliable estimate for the separator. At invariant positions, $d_{\ell,t}(\gamma) = 1$, leaving the base nucleotide distribution unchanged before property guidance.

Auto-balanced property scale. The two forces in Eq. (8), the model prior $\log p_\theta$ and the activity gradient ∇r, live on different scales that drift across reverse steps, so a fixed η_t either lets the property pull dominate (yielding unrealistic sequences) or lets it vanish (yielding low-activity sequences). Following Lee et al. [2023], we set η_t adaptively at each step so that the gradient magnitude matches the prior up to a user ratio c:

$$\eta_t \;=\; \min\Big(\eta_{\max},\; c \cdot \overline{|\log p_\theta|}_{\text{valid}} \,/\, \overline{|\nabla r|}\Big), \tag{10}$$

where the bars denote means over the four nucleotide channels after excluding the DRAKES sentinel values ($\leq -10^5$) at already-unmasked positions, and $\eta_{\max}$ caps the scale when $|\nabla r|$ is small. With c fixed, the property pull stays at a constant fraction of the prior throughout sampling.

To preserve the MDLM unmasking schedule, we add the scaled gradient only to the four nucleotide log-probabilities and renormalise them so that their total mass remains $\Delta_{t\to s}$; the mask channel is left untouched. Property guidance therefore reshapes the categorical distribution *within* nucleotides without altering when MDLM chooses to unmask a position.

5 Experiments

5.1 Experimental Setting

Dataset. We follow the experimental setup of DRAKES [Wang et al., 2025] to ensure direct comparability under an established benchmark protocol, adopting the same chromosomal train/test split. The benchmark is based on the publicly available large-scale enhancer dataset of Gosai et al. [2024], consisting of approximately 700k 200-bp human enhancer sequences with expression measurements quantified in three human cell lines: HepG2, K562, and SK-N-SH.

Generation protocol and evaluation. We follow the DRAKES evaluation scheme [Wang et al., 2025]: each method generates N=640 sequences (20 batches $\times$ 32), which are then re-scored by independent held-out oracle to avoid circularity between guidance and evaluation. This oracle has three MPRA readouts (HepG2, K562, SK-N-SH), is never used for any method's guidance or fine-tuning, and serves as our canonical MPRA-property axis. As a second, MPRA-independent functional readout, we additionally report chromatin accessibility from the binary ATAC classifier [Wang et al., 2025]. For more detailed evaluation protocol, see §S2.

Baselines. We compare LOCOGEN with baselines spanning three categories of regulatory sequence generative models: (1) non-optimizing references: MDLM [Sahoo et al., 2024] (2) Reward-based fine-tuning: DRAKES [Wang et al., 2025], and (3) Training-free guidance: MDLM with classifier guidance (MDLM+CG), SVDD [Li et al., 2024], and DNA-diffusion [DaSilva et al., 2026]. All baselines are run with their authors' released checkpoints and original decoding hyperparameters. We acknowledge concurrent DNA generative models, e.g GLID2E [Cao et al., 2025], Ctrl-DNA [Chen et al., 2025], and GFlowNet [Bengio et al., 2021], but do not include them as baselines since no public checkpoints are available and retraining their upstream models is out of scope. Computation resource requirements are reported in §S1.5.

5.2 Three-axis evaluation framework

We evaluate a generated set $\mathcal{X}^* = \{x^{*(i)}\}_{i=1}^N$ against the training set $\mathcal{X} = \{x^{(j)}\}_{j=1}^M$ along three axes, (1) property, (2) novelty, and (3) compositional plausibility under a fixed protocol decoupled from each method's training or guidance loop. All metrics are reported as mean $\pm$ standard deviation across three sampling seeds.

5.2.1 Property axis.

We assess functional activity using two complementary held-out predictors from Wang et al. [2025]: regulatory potency, measured by target gene expression, and chromatin accessibility, which is required for transcription factors to bind regulatory sequences and modulate gene expression. See §S2.2 for details.

Pred-activity We use an independently released oracle that maps a sequence to a scalar prediction of gene expression, with separate heads for cell lines $c \in \{\mathrm{HepG2}, \mathrm{K562}, \mathrm{SK\text{-}N\text{-}SH}\}$. For each cell line, we report the median predicted activity over generated samples.

ATAC_acc. We use an independently released classifier that maps each sequence to a predicted probability of chromatin accessibility, formulated as a binary classification task distinguishing accessible from inaccessible sequences. We report the mean predicted accessibility over generated samples.

5.2.2 Novelty axis.

Authenticity [Alaa et al., 2022]. We report authenticity score to measure novelty and explicitly diagnose training-set memorization. For each generated sequence $x^{*(i)}$, let $x^{(j_i)}$ be its nearest training neighbour under Euclidean distance $d(\cdot,\cdot)$ in 4-mer frequency space, and let $d_{\mathrm{self}}(x^{(j_i)})$ be that neighbour's leave-one-out 1-NN distance within the training set. We report the fraction of samples that lie farther from their nearest neighbour than the neighbour's natural spacing,

$$\mathrm{Authenticity} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}\Big[d(x^{*(i)}, x^{(j_i)}) > d_{\mathrm{self}}(x^{(j_i)})\Big],$$

with higher values indicating less memorisation.

Rarity [Han et al., 2023]. For each training sample $x^{(j)}$, let $\mathrm{NN}_k(x^{(j)})$ be its distance to its k-th nearest neighbour in 4-mer frequency space, defining a training k-NN ball $B_k(x^{(j)})$ of that radius. The per-sample rarity of an on-manifold generated sequence $x^{*(i)}$ is the radius of the smallest ball containing it,

$$r(x^{*(i)}) = \min_{j\,:\,x^{*(i)} \in B_k(x^{(j)})} \mathrm{NN}_k(x^{(j)}),$$

with off-manifold samples (outside every B_k) excluded so that r isolates on-manifold rarity. To remove dependence on feature scale and sample size, we report r in IQR units relative to the leave-one-out training rarity r^{train}. Thus $\widetilde{r} = 0$ at the training median and $\widetilde{r} = 1$ at p_{75}, with larger values marking samples in rarer training-density regions.

5.2.3 Compositional plausibility axis.

We assess preservation of the natural compositional grammar of regulatory DNA using two sub-metrics, reported as absolute deviations from the training set, $\Delta = |\,\text{method} - \text{train}\,|$. Smaller deviations indicate better preservation.

ΔGini. Motif-usage diversity is measured by the Gini index G over motif counts from JASPAR scans of generated and training sequences [Raluseviciute et al., 2024]. $G = 0$ denotes uniform motif usage, while $G \to 1$ indicates domination by a small subset of motifs. For sorted per-motif counts $f = (f_1, \ldots, f_R)$,

$$G(f) = \frac{1}{R}\left(R + 1 - 2\,\frac{\sum_r (R+1-r)\, f_{(r)}}{\sum_r f_{(r)}}\right) \in [0, 1].$$

See §S2.1 for more details.

Δhits. To measure motif density, we compute hits/seq, the mean number of above-threshold motif occurrences per generated sequence, $\text{hits/seq} = N^{-1}\sum_i h_i$, where h_i is the motif hit count for $x^{*(i)}$. Given hits/seq of natural sequences, deviation in either direction indicates motif under- or over-insertion.

5.3 Diagnostic: two structurally distinct hacking modes

Among baselines, both SVDD and DRAKES successfully push predicted activity upward, but along distinct non-natural directions. Natural enhancers rely on the combinatorial cooperation of multiple TFs, with motifs distributed at a characteristic density. Beyond previously reported metrics, we assess how closely the functional motif composition of generated sequences approaches that of natural regulatory sequences by measuring motif usage diversity and density via ΔGini and Δhits. A large ΔGini indicates that generated sequences are dominated by a few repeated motifs, while a large Δhits indicates stuffed motif within a sequence. In either case, predictor scores can remain high even as the sequence drifts away from natural enhancer composition.

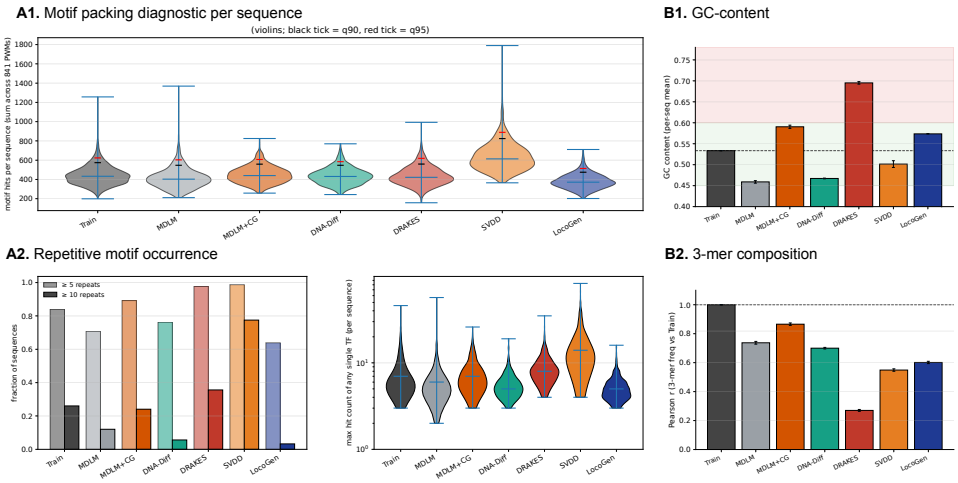

Figure 2: Diagnostics of existing sequence design baselines.

SVDD: motif inflation. In SVDD, motif density per sequence inflates to $1.44\times$ the natural baseline (Δ hits $= 0.248$, from 0.561 to 0.809; Tab. 1, Fig. 2A1), while motif usage concentrates on a narrow subset of the JASPAR repertoire (Δ Gini $= 0.231$, from 0.308 to 0.539; Tab. 1,Fig. 2A2). Repeated use of functionally important motifs is itself biologically plausible; the pathology of SVDD, rather, is that it repeats a single motif up to saturation within a single sequence. Together, these observations suggest that SVDD raises predicted activity by repeatedly stuffing a small set of presumably activity-driving motifs into sequence, rather than by employing the diverse motif vocabulary characteristic of natural enhancers.

DRAKES: GC-rich composition drift. Unlike SVDD, DRAKES does not overpack motifs (Δ hits $= 0.06$; Tab. 1, Fig. 2A1-2), but its motif usage is still skewed toward a narrow subset of the JASPAR repertoire (Δ Gini $= 0.261$; Tab. 1). To understand which motifs DRAKES preferentially favors, we examined the over-represented entries and found that they are predominantly GC-rich (Fig. 2B1). The composition of the generated sequences depart substantially from the natural compositional baseline (Fig. 2B2). Together, these observations suggest that DRAKES inflates predicted activity by drifting toward a generic GC-rich background, rather than by concentrating on the functionally relevant motifs of the target cell line.

5.4 Reaching the empty corner with LOCOGEN

Table 1: Three axis benchmark in HepG2 cell line. 1st, 2nd, and 3rd best results are highlighted.

	Property		Novelty		Plausibility	
Method	Pred-activity ↑	ATAC_acc ↑	Authenticity ↑	Rarity (IQR) ↑	$\lvert\Delta$ Gini$\rvert$ ↓	$\lvert\Delta$ hits$\rvert$ ↓
Train	2.406	0.244	0.728	-	-	-
MDLM	0.157 ± 0.024	0.329 ± 0.198	0.777 ± 0.033	-0.15 ± 0.03	0.051 ± 0.006	0.071 ± 0.001
MDLM+CG	4.516 ± 0.151	0.481 ± 0.022	0.809 ± 0.014	$+0.21 \pm 0.02$	0.169 ± 0.013	0.043 ± 0.007
DNA-Diff	2.595 ± 0.087	0.688 ± 0.293	0.821 ± 0.015	-0.07 ± 0.03	0.002 ± 0.002	0.010 ± 0.002
DRAKES	5.610 ± 0.070	0.804 ± 0.159	0.873 ± 0.004	$+1.08 \pm 0.09$	0.261 ± 0.007	0.060 ± 0.010
SVDD	6.830 ± 0.024	0.442 ± 0.375	0.894 ± 0.008	$+0.84 \pm 0.03$	0.231 ± 0.010	0.248 ± 0.008
LocoGen	4.679 ± 0.021	0.716 ± 0.019	0.906 ± 0.006	$+0.71 \pm 0.00$	0.055 ± 0.005	0.057 ± 0.003

Note. For natural sequences (Train), Δ Gini $= 0.308$ and Δ hits $= 0.561$.

Tables 1, S4, S5, and Figure 1 show that LOCOGEN occupies a distinct region of the three-axis benchmark space. Single-axis optimization leads to clear trade-offs: SVDD and DRAKES obtain the strongest property scores but incur large plausibility shifts, whereas DNA-Diff closely matches the training distribution but yields weaker property optimization and limited rarity. LOCOGEN avoids both failure modes. It achieves competitive property optimization, the best authenticity, positive rarity, and top-three performance on both plausibility metrics. Thus, our claim is not that LOCOGEN is absolutely superior on every metric, but that it is the property-optimizing method that fills the empty corner of the design space: maintaining acceptable performance on property, novelty, and plausibility simultaneously. We further verify that the masked separator learns a meaningful invariant/environmental decomposition rather than an arbitrary partition. The invariant subsequence remains close to the natural baseline, is enriched for activity-associated JASPAR motifs, and has a larger functional impact under ablation than random regions; see §S3.2 for details.

5.4.1 Ablation study

To better understand what drives LOCOGEN's performance, we investigate two complementary questions: (i) Which components of LOCOGEN contribute to its performance? (ii) Can a non-generative approach based on localization and shuffling achieve comparable results?

(i) Table 2 disentangles the contributions of guidance and locality in LOCOGEN. *CG* applies classifier guidance alone ($\gamma = 0$), while *Loc. OOD* (*Learnt*) represents the full model, restricting OOD damping to the mask separator-identified background. *Glob. OOD* control positions at random rather than from the masked separator signal with minimal overlap with the invariant region (Fig. S2). In contrast, *Attack* reverses property guidance at invariant positions, actively disrupting the regions identified by the masked separator. Disabling *CG* substantially reduces both property optimization and novelty while leaving plausibility largely unaffected, highlighting the critical role of classifier guidance in property optimization (rows 1-3 vs. the last row). With *CG* enabled, replacing *Loc. OOD* (*Learnt*) with *Glob. OOD* mildly but consistently degrades property and novelty metrics (row 5 vs. the last row), providing additional evidence that the separator identifies activity-sensitive positions. Meanwhile, the comparison against global OOD damping supports a modest but consistent benefit from localizing OOD control. Removing OOD control substantially degrades all metrics except $\lvert\Delta$hits$\rvert$ (row 4 vs. the last row). Thus, classifier guidance with OOD control helps discover sequences

with stronger properties and greater novelty, while concentrating motif usage on a narrower subset of motifs; the total number of motif hits remains close to the reference distribution. We further examine the effect of OOD control strength γ to assess how strongly to apply distributional calibration during sampling. See Appendix §S3.3.1 for the full results.

Table 2: Locality ablation on HepG2 cell line.

Component			Property		Novelty		Plausibility	
CG	Glob. OOD	Loc. OOD	Pred-activity ↑	ATAC_acc ↑	Authenticity ↑	Rarity (IQR) ↑	$\lvert\Delta$ Gini$\rvert$ ↓	$\lvert\Delta$ hits$\rvert$ ↓
		–	0.157 ± 0.024	0.329 ± 0.198	0.777 ± 0.033	-0.15 ± 0.03	0.051 ± 0.006	0.071 ± 0.001
	✓	–	0.300 ± 0.028	0.314 ± 0.008	0.946 ± 0.002	$+0.46 \pm 0.03$	0.246 ± 0.002	0.167 ± 0.007
		Learnt	0.309 ± 0.020	0.311 ± 0.009	0.947 ± 0.003	$+0.46 \pm 0.03$	0.244 ± 0.004	0.166 ± 0.006
✓		–	4.516 ± 0.151	0.481 ± 0.022	0.809 ± 0.014	$+0.21 \pm 0.02$	0.169 ± 0.013	0.043 ± 0.007
✓	✓	–	4.608 ± 0.061	0.691 ± 0.041	0.887 ± 0.012	$+0.62 \pm 0.11$	0.065 ± 0.021	0.044 ± 0.015
✓		Attack	3.881 ± 0.038	0.656 ± 0.004	0.887 ± 0.012	$+0.64 \pm 0.06$	0.025 ± 0.007	0.083 ± 0.006
✓		Learnt	4.679 ± 0.021	0.716 ± 0.019	0.906 ± 0.006	$+0.71 \pm 0.00$	0.055 ± 0.005	0.057 ± 0.003

(ii) To distinguish the contribution of learned localization from that of generative optimization, we conducted a non-generative control experiment based on a segment-and-shuffle strategy (Tab. S7). See details in §S3.3.2. LearnedMask outperformed heuristic segmentation methods (e.g., predicted activity of 0.442 vs. 0.359 for GC-content masking under uniform shuffling), suggesting that learned localization better identifies activity-relevant regions. Nevertheless, preserving these regions while shuffling the background remained substantially less effective than full LOCOGEN (predicted activity: 0.442 vs. 4.679; ATAC accessibility: 0.280 vs. 0.716). These results suggest that localization alone is insufficient to reproduce LocoGen's performance, highlighting the importance of combining learned localization with a property-guided generative model.

5.5 Robustness to evaluation configuration

To verify that the balanced three-axis profile is not an artifact of our evaluation setup, we evaluated LOCOGEN under (1) independent oracles and (2) a hyperparameter sweep. We first re-evaluated all methods with two publicly available alternative oracles released by prior work [Li et al., 2024, Genentech, 2025]. See §S3.4 for details. Swapping in these oracles leaves most of the ranking intact (Tab. S8,S9,S10). Its three-axis profile is therefore not tied to a specific oracle. Next, we swept LOCOGEN and every baseline over the ranges given in each method's paper or official repository; see §S3.5 for details. Across these sweeps, LOCOGEN retains its balanced three-axis advantage, confirming that the result does not rely on a cherry-picked hyperparameter configuration.

6 Conclusion

We studied sampling-time guidance for regulatory DNA generation and showed that single-axis evaluation can obscure distinct reward-hacking modes. By decomposing evaluation into property, novelty, and plausibility, we identified a desirable region that existing baselines fail to reach. We showed that this gap can be addressed by treating guidance as a problem of **locality** (which positions to modify) rather than scalar strength. Implemented as localized OOD control with an invariant motif mask, this approach improves the three-axis balance on a frozen backbone without retraining. More broadly, our results suggest that locality-aware guidance may be useful in other structured domains with spatially or functionally nonuniform grammars. See §S4 for further discussions on limitations and impact of this work.

Acknowledgments and Disclosure of Funding

We acknowledge core funding from the Wellcome Trust (WT220540/Z/20/A), as well as funding and support from Open Targets.

References

Ahmed Alaa, Boris Van Breugel, Evgeny S Saveliev, and Mihaela Van Der Schaar. How faithful is your synthetic data? sample-level metrics for evaluating and auditing generative models. In *International conference on machine learning*, pages 290–306. PMLR, 2022.

Stephen F Altschul and Bruce W Erickson. Significance of nucleotide sequence alignments: a method for random sequence permutation that preserves dinucleotide and codon usage. *Molecular biology and evolution*, 2(6):526–538, 1985.

Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in neural information processing systems*, 34:17981–17993, 2021.

Emmanuel Bengio, Moksh Jain, Maksym Korablyov, Doina Precup, and Yoshua Bengio. Flow network based generative models for non-iterative diverse candidate generation. *Advances in neural information processing systems*, 34:27381–27394, 2021.

Hanqun Cao, Haosen Shi, Chenyu Wang, Sinno Jialin Pan, and Pheng-Ann Heng. Glid2e: A gradient-free lightweight fine-tune approach for discrete biological sequence design. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Xingyu Chen, Shihao Ma, Runsheng Lin, Jiecong Lin, and Bo Wang. Ctrl-DNA: Controllable cell-type-specific regulatory DNA design via constrained RL. In *Advances in Neural Information Processing Systems*, 2025. Spotlight.

Kevin Clark, Paul Vicol, Kevin Swersky, and David J Fleet. Directly fine-tuning diffusion models on differentiable rewards. In *The Twelfth International Conference on Learning Representations*, 2024.

Lucas Ferreira DaSilva, Simon Senan, Judith F Kribelbauer-Swietek, Zain Munir Patel, Lithin Karmel Louis, Aniketh Janardhan Reddy, Sameer Gabbita, Jonathan D Rosen, Zach Nussbaum, César Miguel Valdez Córdova, et al. Designing synthetic regulatory elements using the generative ai framework dna-diffusion. *Nature Genetics*, 58(1):180–194, 2026.

Oscar Davis, Samuel Kessler, Mircea Petrache, İsmail İ Ceylan, Michael Bronstein, and Avishek J Bose. Fisher flow matching for generative modeling over discrete data. *Advances in Neural Information Processing Systems*, 37:139054–139084, 2024.

Genentech. human-atac-catlas-model. `https://huggingface.co/Genentech/human-atac-catlas-model`, 2025. Enformer fine-tuned with gReLU on the human ATAC CATlas dataset; accessed 2026-10-26.

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, et al. Machine-guided design of cell-type-targeting cis-regulatory elements. *Nature*, 634(8036):1211–1220, 2024.

Jiyeon Han, Hwanil Choi, Yunjey Choi, Junho Kim, Jung-Woo Ha, and Jaesik Choi. Rarity score: A new metric to evaluate the uncommonness of synthesized images. In *International Conference on Learning Representations (ICLR)*. International Conference on Learning Representations, 2023.

Mehran Javanmardi, Mehdi Sajjadi, Ting Liu, and Tolga Tasdizen. Unsupervised total variation loss for semi-supervised deep learning of semantic segmentation. *arXiv preprint arXiv:1605.01368*, 2016.

David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol, and Aaron Courville. Out-of-distribution generalization via risk extrapolation (rex). In *International conference on machine learning*, pages 5815–5826. PMLR, 2021.

Seul Lee, Jaehyeong Jo, and Sung Ju Hwang. Exploring chemical space with score-based out-of-distribution generation. In *International Conference on Machine Learning*, pages 18872–18892. PMLR, 2023.

Haoyang Li, Ziwei Zhang, Xin Wang, and Wenwu Zhu. Learning invariant graph representations for out-of-distribution generalization. *Advances in Neural Information Processing Systems*, 35: 11828–11841, 2022.

Xiner Li, Yulai Zhao, Chenyu Wang, Gabriele Scalia, Gokcen Eraslan, Surag Nair, Tommaso Biancalani, Aviv Regev, Sergey Levine, and Masatoshi Uehara. Derivative-free guidance in continuous and discrete diffusion models with soft value-based decoding. In *Advances in Neural Information Processing Systems*, 2024.

Stuart Lloyd. Least squares quantization in pcm. *IEEE transactions on information theory*, 28(2): 129–137, 1982.

Aaron Lou, Chenlin Meng, and Stefano Ermon. Discrete diffusion modeling by estimating the ratios of the data distribution. In *Proceedings of the 41st International Conference on Machine Learning*, pages 32819–32848, 2024.

Christos Louizos, Max Welling, and Diederik P Kingma. Learning sparse neural networks through l_0 regularization. *arXiv preprint arXiv:1712.01312*, 2017.

Ieva Rauluseviciute, Rafael Riudavets-Puig, Romain Blanc-Mathieu, Jaime A Castro-Mondragon, Katalin Ferenc, Vipin Kumar, Roza Berhanu Lemma, Jeremy Lucas, Jérémy Chèneby, Damir Baranasic, et al. JASPAR 2024: 20th anniversary of the open-access database of transcription factor binding profiles. *Nucleic Acids Research*, 52(D1):D174–D182, 2024.

Kevin Rojas, Ye He, Chieh-Hsin Lai, Yuhta Takida, Yuki Mitsufuji, and Molei Tao. Improving classifier-free guidance in masked diffusion: Low-dim theoretical insights with high-dim impact. In *The Fourteenth International Conference on Learning Representations*, 2026.

Subham S Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T Chiu, Alexander Rush, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. *Advances in Neural Information Processing Systems*, 37:130136–130184, 2024.

Hannes Stark, Bowen Jing, Chenyu Wang, Gabriele Corso, Bonnie Berger, Regina Barzilay, and Tommi Jaakkola. Dirichlet flow matching with applications to dna sequence design. In *International Conference on Machine Learning*, pages 46495–46513. PMLR, 2024.

Sophia Tang, Yinuo Zhang, Alexander Tong, and Pranam Chatterjee. Gumbel-softmax flow matching with straight-through guidance for controllable biological sequence generation. *ArXiv*, pages arXiv–2503, 2025.

Masatoshi Uehara, Yulai Zhao, Chenyu Wang, Xiner Li, Aviv Regev, Sergey Levine, and Tommaso Biancalani. Inference-time alignment in diffusion models with reward-guided generation: Tutorial and review. *arXiv preprint arXiv:2501.09685*, 2025.

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*, 2025.

Kai Zhang, James D Hocker, Michael Miller, Xiaomeng Hou, Joshua Chiou, Olivier B Poirion, Yunjiang Qiu, Yang E Li, Kyle J Gaulton, Allen Wang, et al. A single-cell atlas of chromatin accessibility in the human genome. *Cell*, 184(24):5985–6001, 2021.

Supplementary material

CONTENTS

S1 Supplementary methods **14**

S1.1 Notation . . . 14

S1.2 Reverse process of MDLM . . . 14

S1.3 Masked separator . . . 15

S1.4 Localization-guided sampler . . . 17

S1.5 Computational cost . . . 17

S2 Evaluation protocol **17**

S2.1 Evaluation metric . . . 17

S2.2 Baseline re-scoring and the DRAKES oracle . . . 18

S3 Supplementary results **18**

S3.1 Benchmark analysis on other cell lines. . . . 18

S3.2 Masked separator faithfully disentangles invariant and environmental subsequences. 19

S3.3 Ablation study . . . 19

S3.4 Robust to choice of oracle . . . 20

S3.5 Robust to hyperparameter sweep . . . 21

S4 Discussion **22**

S4.1 Limitations. . . . 22

S4.2 Social Impact . . . 22

S1 Supplementary methods

This section fixes the notation used throughout the paper (§S1.1, Table S1), gives the derivation details of the MDLM reverse step omitted from §3.3 (§S1.2), and then documents the two LOCOGEN modules in the order in which they are introduced in §4: the masked separator (§S1.3) and the localization-guided sampler (§S1.4).

S1.1 Notation

Table S1: Summary of notation. Symbols are grouped by the section in which they are introduced.

Symbol	Description
Sequences and diffusion (§3)	
$\mathcal{A}$	Nucleotide alphabet $\{A, T, G, C\}$
$K = 4$	Alphabet size
L	DNA sequence length ($L = 200$ in all experiments)
$\ell,\ k$	Position index and nucleotide-channel index
$n,\ B$	Sample index within a minibatch and minibatch size
$X \in \mathcal{A}^L$	Discrete DNA sequence
$x_\ell \in \mathcal{A}$	Nucleotide at position ℓ
$\mathbf{x} \in \{0,1\}^{L\times K}$	One-hot encoded DNA sequence; $\mathbf{x}_\ell$ its row at position ℓ
$[\texttt{MASK}],\ \mathbf{m}$	Mask symbol and its one-hot representation in $\{0,1\}^{K+1}$
$\mathbf{z}_t \in \{0,1\}^{L\times(K+1)}$	Masked sequence (diffusion state) at diffusion time t
$s < t$	Next, less-masked diffusion time in the reverse process
α_t	MDLM masking schedule
$\Delta_{t\to s}$	Unmasking mass $(\alpha_s - \alpha_t)/(1 - \alpha_t)$ (Eq. 2)
p_θ	Frozen base (MDLM) generative model
$p_\phi(Y \mid X)$	Auxiliary property predictor
$y^\star$	Target property value
λ	Property guidance strength in the guided distribution
$\mathbf{u}_t,\ f_\phi,\ \beta_t$	MOOD diffusion state, property scorer, and guidance strength
$\gamma \in [0, 1)$	OOD-control strength
Masked separator (§4.1)	
S_ϕ	Separator producing the soft mask
$h_\psi,\ g_\omega$	Invariant predictor and frozen environment encoder
$M \in [0,1]^L$	Soft invariance mask; $M_K = M \otimes \mathbf{1}_K$ its channel-broadcast form
$x^{\text{inv}},\ x^{\text{env}}$	Invariant and environment subsequences (Eq. 5)
$b \in \mathbb{R}^K,\ \bar{b}$	Learnable fill token and its sequence-broadcast form
$E = 3$	Number of auto-discovered environments
$\hat{e}^{(n)},\ R_e,\ c_e$	Environment label, per-environment risk, and centroid
ρ	Target mask density
$\lambda_{\text{inv}}, \lambda_{\text{reg}}, \lambda_{\text{smooth}}$	Loss weights in Eq. 6
Localization-guided sampler (§4.2)	
$r_\xi,\ \nabla r$	Activity guidance model and its gradient w.r.t. $\hat{x}_0$
$\hat{x}_0$	Current x_0-prediction obtained by softmaxing the nucleotide channels
$M_t = S_\phi(\hat{x}_0)$	Mask predicted at diffusion time t
$d_{\ell,t}(\gamma)$	Localized OOD damping factor (Eq. 9)
$\tau_M,\ \tau_{\text{time}}$	Mask and diffusion-time thresholds gating the damping
$\eta_t,\ \eta_{\max},\ c$	Property-pull scale, its cap, and the target magnitude ratio (Eq. 10)

S1.2 Reverse process of MDLM

S1.2.1 Factorization over positions

The per-position transition in Eq. 2 is applied independently at every position, so the reverse transition of the whole sequence factorizes as

$$p_\theta(\mathbf{z}_s \mid \mathbf{z}_t) = \prod_{\ell=1}^{L} p_\theta(\mathbf{z}_{s,\ell} \mid \mathbf{z}_t). \tag{S1}$$

S1.2.2 Point mass at already-decoded positions

At diffusion time t, let $a_{t,\ell} \in \mathcal{A}_{\text{mask}}$ denote the categorical token at position ℓ, with one-hot representation

$$\mathbf{z}_{t,\ell} = \text{onehot}(a_{t,\ell}) \in \{0,1\}^5, \qquad \sum_{k=1}^{5} [\mathbf{z}_{t,\ell}]_k = 1. \tag{S2}$$

The first branch of Eq. 2 is then the point mass

$$\delta_{\mathbf{z}_{t,\ell}}(\mathbf{z}_{s,\ell}) = \begin{cases} 1, & \mathbf{z}_{s,\ell} = \mathbf{z}_{t,\ell}, \\ 0, & \text{otherwise}, \end{cases} \tag{S3}$$

which freezes every position whose token has already been decoded; only positions still carrying $\mathbf{m}$ are resampled. This is the SUBS parameterization that LOCOGEN inherits, and it is the reason the damping in §S1.4 can act only on still-masked positions.

S1.3 Masked separator

S1.3.1 Backbone architecture

Each of the three networks S_ϕ, h_ψ, and g_ω is a stack of dilated 1D convolutions of identical capacity to the DRAKES classifier [Wang et al., 2025], with the hyperparameters of Table S2: 5 stacks of 5 dilations with rates $\{1, 1, 4, 16, 64\}$, kernel size 9, hidden width $H = 128$, residual connections within each stack, and ReLU activations. The input is the one-hot tensor $\mathbf{x} \in \{0,1\}^{L \times K}$ with $L = 200$ and $K = 4$. The separator head applies a position-wise sigmoid to produce $M \in [0,1]^L$, with its output bias initialized at $\text{logit}(\rho)$ so that $\mathbb{E}[M] \approx \rho$ at the start of training. The predictor reads out the invariant subsequence through a 2-layer head on global mean pooling,

$$\hat{y}^{\text{inv}} \;=\; W_2 \,\text{ReLU}\Big(W_1 \,\tfrac{1}{L} \textstyle\sum_{\ell=1}^{L} h_\psi(x^{\text{inv}})_\ell\Big), \qquad W_1 \in \mathbb{R}^{H \times H},\; W_2 \in \mathbb{R}^{1 \times H}, \tag{S4}$$

while the environment encoder produces a single embedding per sequence,

$$g_\omega(x^{\text{env}}) \;=\; \tfrac{1}{L} \textstyle\sum_{\ell=1}^{L} \tilde{g}_\omega(x^{\text{env}})_\ell \;\in\; \mathbb{R}^H, \tag{S5}$$

where $\tilde{g}_\omega$ denotes the convolutional trunk before pooling. We write $g_\omega(x^{\text{env}})$ for the pooled embedding throughout, as in §4.1.

The three networks share architecture but *not* weights, and only S_ϕ and h_ψ are trained: g_ω is randomly initialized and kept frozen. A trainable environment encoder can lower $\mathcal{L}_{\text{inv}}$ by collapsing the embedding space itself rather than by improving the partition; tying g_ω to h_ψ has the same effect, since the predictor's collapse attractor ($x^{\text{inv}} \to \bar{b}$, $\hat{y}^{\text{inv}}$ constant) would then corrupt the embedding space used for k-means (§S1.3.3). Freezing g_ω leaves the separator as the only route by which the environment partition can change, which is what makes the V-REx penalty informative about the mask.

S1.3.2 Auxiliary loss terms

Mask anchoring ($\mathcal{L}_{\text{reg}}$). A symmetric quadratic penalty pulling the average mask density to the target ρ, with an optional bimodal sub-term:

$$\mathcal{L}_{\text{reg}} \;=\; (\bar{M} - \rho)^2 \;+\; \lambda_{\text{bim}}\, \overline{M(1-M)}, \qquad \bar{M} \;=\; \tfrac{1}{BL} \textstyle\sum_{n,\ell} M_\ell^{(n)}. \tag{S6}$$

The first term is expected-L_0 anchoring [Louizos et al., 2017]. We set $\lambda_{\text{bim}} = 0$: with $\lambda_{\text{bim}} > 0$ the $\{0,1\}$ valleys become attractors and combine with the mean term to collapse the separator within the first epoch. The target density ρ is held as a *frozen buffer* rather than a trainable parameter; when made learnable it gets pulled toward $\bar{M}$, which hollows out the penalty whenever the mask collapses.

Spatial smoothness ($\mathcal{L}_{\text{smooth}}$). A fused-lasso total-variation penalty on adjacent mask values [Javanmardi et al., 2016]:

$$\mathcal{L}_{\text{smooth}} \;=\; \tfrac{1}{B(L-1)} \textstyle\sum_{n} \sum_{\ell=1}^{L-1} \big|M_{\ell+1}^{(n)} - M_\ell^{(n)}\big|. \tag{S7}$$

This encourages contiguous selected blocks rather than scattered single positions, consistent with the cis-regulatory grammar in which functional elements occupy short contiguous motifs.

V-REx variance estimator. $\mathcal{L}_{\text{inv}}$ in Eq. 7 is computed in its biased form. Environments containing fewer than two samples are dropped from the variance estimate; if fewer than two environments survive, we set $\mathcal{L}_{\text{inv}} = 0$ for that batch.

S1.3.3 Environment inference by k-means

Because x^{env} is defined by the current mask, the inferred environments shift as S_ϕ is updated. We therefore refit k-means at the start of every epoch, so that the V-REx penalty is always measured against the partition induced by the current mask, and hold the labels fixed within the epoch, which prevents the separator from lowering $\text{Var}_e[R_e]$ by reshuffling the environments themselves (the standard GIL alternating optimization [Li et al., 2022]). Writing j for the epoch index, one round proceeds as follows.

Step 1 — Snapshot collection. At the start of each epoch $j \geq t_{\text{warmup}}$, a copy Θ_j of the current parameters is frozen; since g_ω is already frozen, the snapshot pins the separator, and hence the mask that defines x^{env}. Up to $N_{\text{km}} = 8192$ samples are drawn from the train loader and their environment embeddings $g_\omega(x^{\text{env}}; \Theta_j)$ are collected. For this pass alone we use *hard-input* mixing: the fill token b is **not** injected, so k-means clusters on actual background nucleotide content rather than on a b-contaminated mixture.

Step 2 — k-means fit. Lloyd's algorithm [Lloyd, 1982] is run with $E = 3$ clusters and $T_{\text{km}} = 20$ iterations, initialized by random-sample seeding indexed by the epoch number. The resulting centroids $\{c_e\}_{e=1}^{E} \subset \mathbb{R}^H$ are cached.

Step 3 — Minibatch assignment. During each training step within the epoch, samples are assigned to the nearest cached centroid under squared ℓ_2 distance:

$$\hat{e}^{(n)} \;=\; \arg\min_{e \in \{1,\dots,E\}} \left\| g_\omega(x^{\text{env}(n)}; \Theta_j) - c_e \right\|_2^2. \tag{S8}$$

These labels feed $\mathcal{L}_{\text{inv}}$ (Eq. 7). Because both the centroids and the embeddings are taken from the same frozen snapshot Θ_j, environment labels and centroids stay mutually consistent for the whole epoch.

Warmup. For the first $t_{\text{warmup}} = 1$ epoch no centroids exist; we set $\mathcal{L}_{\text{inv}} = 0$ and train as pure ERM on x^{inv}. Subsequently the refit runs at the start of every epoch.

Persistence. The final centroids $\{c_e\}$ are saved with the model checkpoint for reproducibility and for the environment diagnostics of §S3.2. The sampler of §S1.4 consumes only the separator output $M_t = S_\phi(\hat{x}_0)$ and does not consult the centroids.

Table S2: Hyperparameters for the masked separator (§S1.3).

Symbol	Value	Description	Source
H	128	hidden dim of CNN backbones	`model.hidden_dim`
stacks	5	dilated-CNN stack depth	`model.num_cnn_stacks`
dropout	0.1	per-conv dropout	`model.dropout`
ρ	0.3	target invariant mask density	`init_rho`
λ_{inv}	1.0	V-REx weight	config
λ_{reg}	5.0	mask-anchoring weight	config
λ_{smooth}	0.1	fused-lasso TV weight	config
λ_{bim}	0.0	bimodal sub-term weight (disabled)	config
E	3	number of k-means environments	`num_envs`
T_{km}	20	k-means Lloyd iterations	`kmeans_iters`
N_{km}	8192	k-means subsample cap	`kmeans_samples`
t_{warmup}	1 epoch	ERM-only warmup	`warmup_epochs`
optimiser	Adam	lr 10^{-4}, grad-clip max-norm 1.0	`train.lr`

S1.4 Localization-guided sampler

The sampler operates entirely at inference time on frozen components: the MDLM prior p_θ, the separator S_ϕ, and the activity guidance model r_ξ. At each reverse step $t \to s$ it evaluates p_θ once, forms the x_0-prediction $\hat{x}_0$, queries the separator to obtain $M_t = S_\phi(\hat{x}_0)$, and combines the three factors of Eq. 8: the localized damping $d_{\ell,t}(\gamma)$ of Eq. 9, the MDLM unmasking mass $\Delta_{t\to s}$, and the auto-balanced property pull $\eta_t \nabla r$ of Eq. 10. Damping is gated by two thresholds: it acts only where $M_{t,\ell} < \tau_M$, i.e. on environment positions, and only for $t < \tau_{\text{time}}$, i.e. in the second half of sampling, where $\hat{x}_0$ is informative enough for S_ϕ to classify positions reliably. Because already-decoded positions are frozen by the point mass of §S1.2.2, damping is in effect restricted to still-masked positions.

Sentinel-aware normalization. The auto-balance of Eq. 10 normalizes the property pull by the mean magnitude of the prior log-probabilities. DRAKES [Wang et al., 2025] writes a large negative sentinel at positions whose token has already been determined, in order to block re-sampling. Taken without filtering, these sentinel entries dominate $\overline{|\log p_\theta|}$, inflate the ratio, and pin η_t at its cap $\eta_{\max}$ at every step, which silently disables auto-balance. We therefore restrict the mean to non-sentinel entries ($\log p_\theta > -10^5$). The same correction is applied when running the classifier-guided baselines, so that no method is penalized by this implementation detail.

Finalization. Tokens are drawn from the renormalized categorical distribution over the four nucleotide channels, with the copy rule of MDLM keeping already unmasked positions fixed. After the final reverse step, one additional forward pass with per-position $\arg\max$ removes any residual mask tokens, as in DRAKES [Wang et al., 2025].

Remaining architecture and configuration details are provided in the code; hyperparameter settings are summarized in Table S3.

Table S3: Hyperparameters for the localization-guided sampler (§S1.4).

Symbol	Value	Description
γ	0.7	OOD damping strength ($d_{\ell,t} = 1 - \sqrt{\gamma}$, Eq. 9)
τ_M	0.5	mask threshold; damping acts where $M_{t,\ell} < \tau_M$
τ_{time}	0.5	damping time threshold (active only for $t < \tau_{\text{time}}$)
c	3.0	target property-pull-to-prior magnitude ratio (Eq. 10)
$\eta_{\max}$	20	cap on η_t when $\lvert\nabla r\rvert$ is small
sentinel cutoff	-10^5	threshold below which $\log p_\theta$ is excluded from the norm
batch / seeds	32 / 3	sampling batch size and number of seeds
N	640	sequences per method per cell line per seed (20×32)

S1.5 Computational cost

All experiments were run on a single NVIDIA H200 GPU per job, with 16 CPU cores and 40 GB of host RAM. Training the masked separator required approximately 5.5 hours per epoch, with peak host-side RAM usage of 4.6 GB. For sequence generation, sampling 640 sequences with 128 reverse-diffusion steps took 191 wall-clock seconds per random seed, with peak host-side RAM usage of 3.5 GB.

S2 Evaluation protocol

This section presents further details omitted from the main text due to space constraints.

S2.1 Evaluation metric

ΔGini. We scan generated/training sequences against the JASPAR motif database TF position-specific scoring matrices [Rauluseviciute et al., 2024], thresholded at false positive rate 10^{-3} under a uniform ACGT background. Following Gosai et al. [2024], motifs are restricted to those whose

constituent TF subunits are all expressed at TPM > 1 in the target cell type. The Gini coefficient over per-motif counts $f = (f_1, \ldots, f_R)$ sorted by ascending order is

$$G(f) = \frac{1}{R}\left(R + 1 - 2\,\frac{\sum_r (R+1-r)\, f_{(r)}}{\sum_r f_{(r)}}\right) \in [0, 1].$$

S2.2 Baseline re-scoring and the DRAKES oracle

To avoid circularity from using the same predictor for both classifier-guided generation and evaluation, we used separate independent oracles for generation guidance and sequence evaluation, following the DRAKES protocol [Wang et al., 2025]. Specifically, `reward_oracle_ft.ckpt` is used only for guidance or reward-based fine-tuning, whereas evaluation is performed using the separately distributed `reward_oracle_eval.ckpt`. This held-out evaluator is a gReLU `EnformerPretrainedModel` with three MPRA readouts corresponding to HepG2, K562, and SK-N-SH, and never contributes to the guidance or fine-tuning signal of any method. As a second functional readout independent of MPRA, we additionally report chromatin accessibility from the binary ATAC classifier `binary_atac_cell_lines.ckpt` [Wang et al., 2025]; the two predictors are trained independently.

S3 Supplementary results

S3.1 Benchmark analysis on other cell lines.

Table S4: Three axis benchmark in K562 cell line (n=640 $\times$ 3 seeds, mean $\pm$ std). 1st, 2nd, and 3rd best results are highlighted.

	Property		Novelty		Plausibility	
Method	Pred-activity↑	ATAC_acc ↑	Authenticity ↑	Rarity (IQR) ↑	$\lvert\Delta$ Gini$\rvert$ ↓	$\lvert\Delta$ hits$\rvert$ ↓
Train	2.700	0.281	0.717	-	-	-
MDLM	0.106 ± 0.028	0.367 ± 0.227	0.777 ± 0.033	-0.03 ± 0.03	0.049 ± 0.005	0.067 ± 0.001
MDLM+CG	6.433 ± 0.052	0.852 ± 0.196	0.810 ± 0.006	$+0.41 \pm 0.03$	0.260 ± 0.007	0.093 ± 0.005
DNA-Diff	2.979 ± 0.046	0.479 ± 0.311	0.841 ± 0.007	$+0.13 \pm 0.03$	0.028 ± 0.020	0.038 ± 0.008
DRAKES	6.493 ± 0.155	0.863 ± 0.140	0.873 ± 0.004	$+1.07 \pm 0.08$	0.251 ± 0.008	0.069 ± 0.011
SVDD	9.479 ± 0.055	0.928 ± 0.012	0.848 ± 0.031	$+0.77 \pm 0.03$	0.305 ± 0.002	0.241 ± 0.005
LocoGen	6.285 ± 0.064	0.900 ± 0.012	0.893 ± 0.010	$+0.77 \pm 0.01$	0.094 ± 0.003	0.033 ± 0.002

Note. For natural sequences (Train), Δ Gini $= 0.312$ and Δ hits $= 0.555$

Table S5: Three axis benchmark in SK-N-SH cell line (n=640 $\times$ 3 seeds, mean $\pm$ std). 1st, 2nd, and 3rd best results are highlighted.

	Property		Novelty		Plausibility	
Method	Pred-activity ↑	ATAC_acc ↑	Authenticity ↑	Rarity (IQR) ↑	$\lvert\Delta$ Gini$\rvert$ ↓	$\lvert\Delta$ hits$\rvert$ ↓
Train	2.376	0.212	0.713	-	-	-
MDLM	0.112 ± 0.026	0.391 ± 0.217	0.777 ± 0.033	-0.09 ± 0.03	0.071 ± 0.006	0.080 ± 0.001
MDLM+CG	5.047 ± 0.101	0.429 ± 0.295	0.828 ± 0.010	$+0.23 \pm 0.06$	0.120 ± 0.003	0.007 ± 0.005
DNA-Diff†	0.033 ± 0.015	0.079 ± 0.087	0.821 ± 0.015	$+0.09 \pm 0.03$	0.024 ± 0.000	0.034 ± 0.000
DRAKES	6.992 ± 0.169	0.879 ± 0.145	0.873 ± 0.004	$+1.07 \pm 0.09$	0.241 ± 0.008	0.095 ± 0.011
SVDD	7.989 ± 0.041	0.356 ± 0.025	0.868 ± 0.021	$+1.25 \pm 0.06$	0.260 ± 0.001	0.213 ± 0.004
LocoGen	5.304 ± 0.186	0.747 ± 0.048	0.897 ± 0.016	$+0.69 \pm 0.06$	0.012 ± 0.006	0.081 ± 0.004

Note. For natural sequences (train set), Δ Gini $= 0.321$ and Δ hits $= 0.572$. †:DNA-Diff [DaSilva et al., 2026] trained on GM12878, HepG2, hESCT0, K562; SK-N-SH not in label set. K562-anchored seqs scored under SK-N-SH oracle head.

S3.2 Masked separator faithfully disentangles invariant and environmental subsequences.

We next asked whether the masked separator module recovers a meaningful invariant/environmental decomposition. Three lines of evidence support that it does (Fig. S1). First, the three environmental subsequences exhibit clearly distinct GC-content distributions while the invariant subsequence sits near the natural baseline (Fig. S1A), and the same separation holds in 3-mer PCA space, where each environment forms a distinct cluster around its own centroid (Fig. S1B). Second, the invariant subsequence is enriched for canonical activity-driving JASPAR motifs, predominantly ETS-family factors (Fig. S1C). Third, ablating only the invariant subsequence causes a substantially larger drop in oracle-predicted activity than ablating an equally-sized random region (Fig. S1D), showing that the invariant region is functionally responsible for the predicted activity rather than being an arbitrary partition.

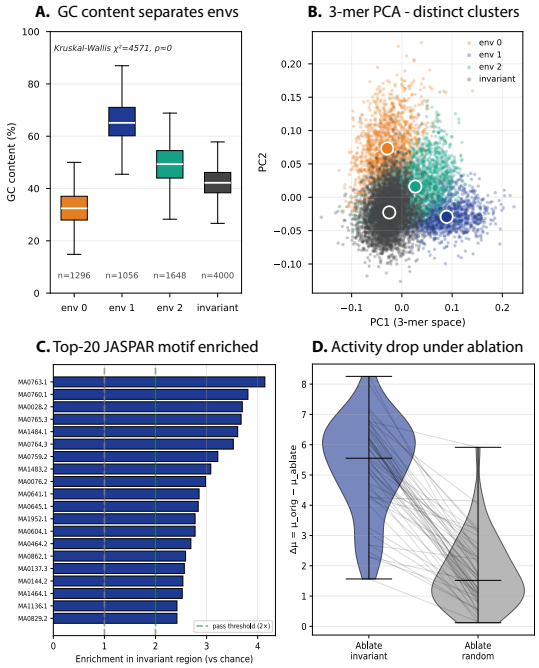

Figure S1: Sequence-level invariance learning recovers regulatory grammar.

S3.3 Ablation study

To better interpret the global-OOD setting in §5.4.1, we conducted an additional experiment in which the same number of masked positions was distributed uniformly at random across the sequence (Fig. S2). We confirmed that these randomly selected positions exhibited minimal overlap with the invariant region.

S3.3.1 Effect of OOD control strength

Table S6 studies the effect of the OOD control strength γ. Without OOD control the model achieves reasonable predicted activity but shows weak ATAC accessibility, low authenticity, limited rarity, and a large deviation in $|\Delta\text{Gini}|$. Increasing γ to 0.3 substantially improves ATAC accessibility, authenticity, rarity, and both plausibility metrics, indicating that moderate OOD control leads to generate more plausible sequences. At $\gamma = 0.7$, LOCOGEN achieves the best authenticity, rarity, and $|\Delta\text{Gini}|$, while also maintaining strong predicted activity and the highest ATAC accessibility. Although $\gamma = 0.3$ gives slightly higher *pred-activity* and the lowest $|\Delta\text{hits}|$, the $\gamma = 0.7$ setting provides the most balanced operating point across property, novelty, and plausibility. These results support the role of OOD control as a calibration mechanism that controls the trade-off between aggressive property optimization and distributional realism.



Figure S2: **The learnt invariant region is not a random subset of positions.** (A) For ten HepG2 enhancers, the learnt invariant mask (dark) and a count-matched random mask (purple) along the 200bp window; the light band is the OOD-damped background. Both masks select the same number of positions, so any difference in the ablation is attributable to which positions are spared rather than how many. (B) Per-sequence Jaccard index between the two masks.

Table S6: OOD strength sweep on HepG2.

γ	Pred-activity ↑	ATAC$_{acc}$ ↑	Auth.↑	Rarity (IQR)↑	$\lvert\Delta$Gini$\rvert$ ↓	$\lvert\Delta$hits$\rvert$ ↓
0.0	4.516 ± 0.151	0.481 ± 0.022	0.809 ± 0.011	$+0.22 \pm 0.01$	0.169 ± 0.010	0.043 ± 0.005
0.3	4.738 ± 0.061	0.651 ± 0.007	0.870 ± 0.004	$+0.53 \pm 0.03$	0.095 ± 0.003	0.027 ± 0.002
0.7	4.679 ± 0.0	0.716 ± 0.019	0.906 ± 0.005	$+0.71 \pm 0.00$	0.055 ± 0.004	0.057 ± 0.002

S3.3.2 Segment and shuffle baseline

We uniformly sampled regulatory sequences from the training set, preserved positions designated as foreground, and shuffled the remaining background positions. Here, foreground and background refer to regions selected for preservation and modification, respectively, rather than the invariant and environment components defined in LocoGen. We evaluated combinations of three foreground segmentation methods × two background replacement strategies.

- **Foreground segmentation**
  - Position-specific scoring matrix (PSSM) motif scan
  - GC content-based segmentation
  - Predictor saliency
- **Background replacement**
  - Uniform random replacement
  - Altschul-Erikson dinucleotide shuffle (i.e., dinucleotide-preserving) [Altschul and Erickson, 1985]

S3.4 Robust to choice of oracle

To demonstrate the generalizability of LocoGen across different experimental settings, we evaluated robustness to different oracles for activity and accessibility evaluation. We evaluated our method using two publicly available alternative oracles released by prior work [Li et al., 2024, Genentech, 2025].

Table S7: Comparison with non-generative segment-and-shuffle controls on HepG2.

Segmentation	Replacement	Property		Novelty		Plausibility	
		Pred-act. ↑	ATAC_{acc} ↑	Authent. ↑	Rarity (IQR) ↑	$\lvert\Delta\text{Gini}\rvert$ ↓	$\lvert\Delta\text{hits}\rvert$ ↓
JASPAR PSSM	Uniform random	0.275	0.206	0.928	+0.25	0.155	0.114
JASPAR PSSM	Dinuc.-preserving	0.122	0.092	0.815	−0.12	0.069	0.066
GC content	Uniform random	0.359	0.252	0.907	+0.13	0.148	0.129
GC content	Dinuc.-preserving	0.078	0.086	0.809	−0.16	0.054	0.076
Saliency	Uniform random	0.288	0.249	0.961	+0.30	0.167	0.138
Saliency	Dinuc.-preserving	0.199	0.129	0.840	−0.03	0.101	0.095
LearnedMask	Uniform random	0.442	0.280	0.955	+0.38	0.166	0.171
LearnedMask	Dinuc.-preserving	0.287	0.109	0.823	−0.05	0.067	0.113
LocoGen		4.679	0.716	0.906	+0.71	0.055	0.057

- Activity oracle [Li et al., 2024]: an independent Enformer-based predictor trained on the same Gosai dataset, but with a different train/test split and substantially more training data than the DRAKES oracle.
- Accessibility oracle [Genentech, 2025]: an independent Enformer-based predictor trained on the Human ATAC Atlas [Zhang et al., 2021], a different dataset for chromatin accessibility prediction.

Overall, LOCOGEN consistently maintains strong property performance across independent evaluation oracles, demonstrating that our conclusions are not tied to a specific oracle.

Table S8: **Robustness to oracle choice (HepG2).** Beyond the DRAKES evaluation oracles used in the main text, we add two predictors that differ in training split and in data source, respectively. Rank across the six methods in parentheses; values are mean ± s.d. where seed replicates are available.

Method	Activity		Accessibility	
	DRAKES Pred-activity	SVDD DNA_evaluation *alternative training split*	DRAKES ATAC_acc	human-atac-catlas *different data source*
MDLM	0.157 (6)	0.408 ± 0.013 (6)	0.329 (6)	0.050 ± 0.009 (6)
MDLM+CG	4.516 (4)	4.434 ± 0.097 (4)	0.481 (4)	0.459 ± 0.014 (5)
DNA-Diff	2.595 (5)	2.721 ± 0.046 (5)	0.688 (3)	0.517 ± 0.020 (4)
DRAKES	5.610 (2)	5.135 ± 0.076 (2)	0.804 (1)	0.829 ± 0.006 (1)
SVDD	6.830 (1)	6.187 ± 0.018 (1)	0.442 (5)	0.547 ± 0.010 (3)
LocoGen	**4.679** (3)	$\mathbf{4.660 \pm 0.011}$ (3)	**0.716** (2)	$\mathbf{0.729 \pm 0.002}$ (2)

Table S9: **Robustness to oracle choice (K562).** Same four oracles and conventions as Table S8; rank across the six methods in parentheses, values are mean ± s.d. where seed replicates are available.

Method	Activity		Accessibility	
	DRAKES Pred-activity	SVDD DNA_evaluation *alternative training split*	DRAKES ATAC_acc	human-atac-catlas *different data source*
MDLM	0.106 (6)	0.381 ± 0.031 (6)	0.367 (6)	0.055 ± 0.006 (6)
MDLM+CG	6.433 (3)	6.191 ± 0.042 (2)	0.852 (4)	0.688 ± 0.022 (4)
DNA-Diff	2.979 (5)	3.038 ± 0.036 (5)	0.479 (5)	0.359 ± 0.009 (5)
DRAKES	6.493 (2)	6.031 ± 0.114 (4)	0.863 (3)	0.837 ± 0.005 (3)
SVDD	9.479 (1)	8.490 ± 0.064 (1)	0.928 (1)	0.899 ± 0.005 (1)
LocoGen	**6.285** (4)	$\mathbf{6.085 \pm 0.014}$ (3)	**0.900** (2)	$\mathbf{0.846 \pm 0.002}$ (2)

S3.5 Robust to hyperparameter sweep

We ran a hyperparameter sweep of both our method and the baselines, sweeping each over the range specified in its paper or official repository. Across these sweeps, LOCOGEN retains its balanced three-axis advantage, confirming that the result does not rely on a cherry-picked configuration.

Table S10: **Robustness to oracle choice (SK-N-SH).** Same four oracles and conventions as Table S8; rank across the six methods in parentheses, values are mean ± s.d. where seed replicates are available.

Method	Activity		Accessibility	
	DRAKES Pred-activity	SVDD DNA_evaluation *alternative training split*	DRAKES ATAC_acc	human-atac-catlas *different data source*
MDLM	0.112 (5)	0.340 ± 0.033 (6)	0.391 (4)	0.030 ± 0.005 (6)
MDLM+CG	5.047 (4)	4.970 ± 0.104 (4)	0.429 (3)	0.377 ± 0.020 (3)
DNA-Diff	0.033 (6)	0.720 ± 0.021 (5)	0.079 (6)	0.040 ± 0.002 (5)
DRAKES	6.992 (2)	6.132 ± 0.153 (2)	0.879 (1)	0.839 ± 0.007 (1)
SVDD	7.989 (1)	6.714 ± 0.103 (1)	0.356 (5)	0.192 ± 0.004 (4)
LocoGen	**5.304** (3)	$\mathbf{5.294 \pm 0.156}$ (3)	**0.747** (2)	$\mathbf{0.629 \pm 0.020}$ (2)

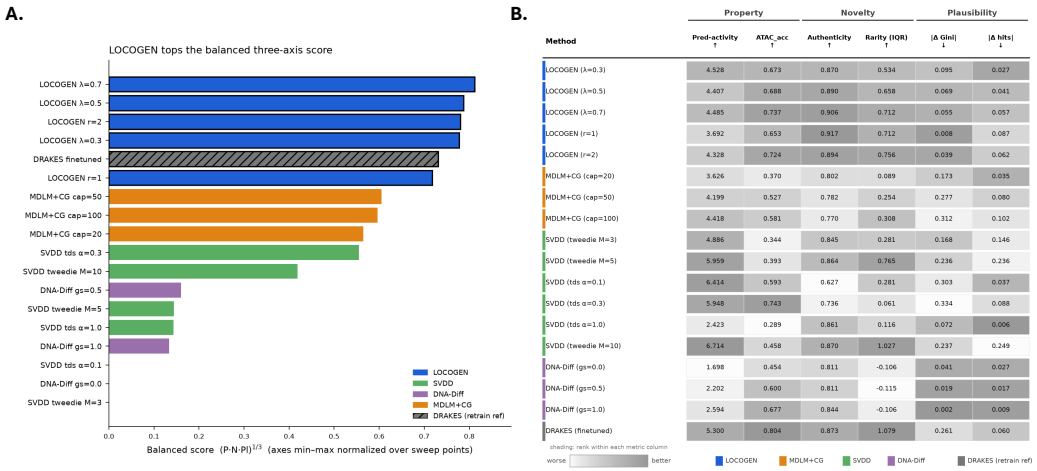

Figure S3: (A). Balanced score over three evaluation axis where LocoGen achieves the highest balanced score across sweep points. (B) Evaluation metrics across the sweep, with one row per hyperparameter setting. The value in parentheses indicates each method's strength setting: γ denotes OOD/calibration strength (`ood_lambda`); r denotes the property-pull ratio (`alpha_ratio`) at $\gamma = 0.7$ (the paper default $r = 3.0$ corresponds to the $\gamma = 0.7$ row, while $r = 1, 2$ are lower-pull variants); cap denotes the MDLM+CG classifier-guidance α-cap (`alpha_cap`); M denotes the number of SVDD candidates per step (Tweedie/SVDD-PM); α_{TDS} denotes the SVDD-TDS importance-resampling temperature; and gs denotes the DNA-Diff classifier-free guidance scale (`cond_weight`; author default 1.0). DRAKES fine-tunes model weights for a fixed reward rather than steering generation at inference time, and is therefore included as a retraining reference rather than a directly comparable operating point.

S4 Discussion

S4.1 Limitations.

Our experiments are conducted on a single backbone, MDLM, to isolate the effect of localized OOD guidance from backbone-dependent variation. Validation on alternative backbones, such as flow matching or autoregressive models, and a full comparison across backbones are left to future work. Computational evaluation alone, while standard in this domain, cannot substitute for *in vivo* experimental validation of generated sequences, a precondition for treating any sampling-time guidance method as practically useful.

S4.2 Social Impact

Our work may contribute to bioengineering by improving the ability to generate novel biological sequences beyond the training distribution. Such capability can expand the searchable design space

and accelerate the discovery of sequences with useful properties. In practical settings, this could reduce the cost and time required for experimental screening, while supporting applications in therapeutic development, enzyme engineering, antibody design, and synthetic biology.

However, the broader impact of biological sequence generation depends on how the technology is used. Since generative models can produce sequences that differ substantially from naturally observed examples, they may also introduce biosafety and dual-use risks. For example, the same methods that help design beneficial proteins or regulatory elements could potentially be misused to explore harmful biological functions. Therefore, responsible use requires careful consideration of dataset provenance, model access, downstream experimental validation, and risk assessment tailored to the intended application.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes], [No], or [N/A].
- [N/A] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for [N/A]).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will also be asked to include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While [Yes] is generally preferable to [No], it is perfectly acceptable to answer [No] provided a proper justification is given (e.g., error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering [No] or [N/A] is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

1. **Claims**

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

   Answer: [Yes]

   Justification: I double checked whether the main claim reflects the paper's contribution and scope.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: I discussed limitations of this work in Section S4.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.

- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

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

   Answer: [N/A]

   Justification: The paper does not make formal theoretical claims or introduce theorems, lemmas, or propositions. Instead, the contributions are methodological and empirical analyses.

   Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: We used public benchmark dataset and codes are released in github repo.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.

- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

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

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

   Answer: [Yes]

   Justification: We used public benchmark dataset and codes are released in github repo.

   Guidelines:

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

6. **Experimental setting/details**

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

Answer: [Yes]

Justification:

Guidelines:

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

7. **Experiment statistical significance**

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

Answer: [Yes]

Justification: How we depicted error bars and standard deviation are disclosed in figure legend.

Guidelines:

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

8. **Experiments compute resources**

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

Answer: [Yes]

Justification: Compute resources are described in §S1.5.

Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: We followed the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: Societal impacts are discussed in §S4.

    Guidelines:

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

11. **Safeguards**

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

    Answer: [N/A]

    Justification: Given the early-stage nature of this research, the risk of misuse is currently limited.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.

- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: We only used software licensed under the MIT License for non-commercial purposes.

    Guidelines:

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

13. **New assets**

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

    Answer: [Yes]

    Justification: New assets introduced in the paper would be intellectual assets, mainly codes. README files for full instruction is provided.

    Guidelines:

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

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

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

    Answer: [N/A]

    Justification: This paper does not include any human subjects nor crowdsourcing.

    Guidelines:

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

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

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

    Answer: [N/A]

    Justification: This paper does not include any human subjects.

    Guidelines:

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

16. **Declaration of LLM usage**

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

    Answer: [N/A]

    Justification: LLMs were used only in a supportive role for writing, editing, formatting, and code assistance.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.