Elucidating the Design Space of Generative Models for Single-Cell Perturbation Prediction

Sanjukta Bhattacharya [1] **Christian Gensbigler** [2] **Shaamil Karim** [2]

Abstract

Next-token prediction has produced predictable scaling in language, but the recipe presumes a sequence of tokens with a meaningful order. Single-cell RNA-seq counts have no natural gene ordering, so applying the recipe directly to raw expression fails under an ill-suited left-to-right bias. We instead ask whether a learned latent can supply the structure the recipe needs. We introduce `ExpressionVAE` (eVAE), a discrete-latent perturbation model that compresses each cell into a short sequence of discrete codes through a finite-scalar-quantization (FSQ) bottleneck and trains a perturbation-conditioned discrete prior over those codes. On Replogle and Parse 1M, eVAE sets a new state of the art on every distributional metric and leads on most cell-eval perturbation metrics, with Fréchet distance and MMD^2 roughly 3 to $20\times$ lower than the strongest continuous-latent baseline. Swapping the prior between autoregressive and masked discrete diffusion leaves performance near-identical, isolating the gain to the discrete latent itself rather than the prior family. A decoder-head ablation then exposes a single design axis, the richness of the predictive distribution at inference, that splits the standard metrics into two groups, variance-sensitive and mean-sensitive, which move in opposite directions along the axis. Finally, on a held-out CRISPRi reversion benchmark of 1,732 perturbations under inflammatory cytokine stress, the frozen eVAE encoder outperforms UMAP and differential expression and matches scGPT on perturbation ranking at a fraction of the data.[1] We release our code.[2]

[1]University of Bristol, Bristol, United Kingdom [2]Dartmouth College, Hanover, NH, USA. Correspondence to: Sanjukta Bhattacharya <ae25872@bristol.ac.uk>.

Accepted at the 2026 Workshop on Generative and Agentic AI for Biology (ICML 2026)

[1]https://sanjukta7.github.io/sclatents/

[2]https://github.com/sanjukta7/sc-evae

1. Introduction

A cell can be viewed as an information-processing system that maps environmental inputs to molecular responses through an internal regulatory program (Fitch, 2021). While the structure of the regulatory network itself is not directly accessible, single-cell RNA sequencing provides a high-dimensional partial observation of its activity (Macosko et al., 2015; Gulati et al., 2020; Zeng and Dai, 2019). Crucially, this observation is discrete: we do not measure continuous expression levels, but count individual molecular events. Across many cells and conditions, these observations implicitly encode the statistical dependencies induced by the underlying regulatory structure. The objective of a virtual cell model is to learn these dependencies well enough to generate realistic cellular states and to predict how modifications to the regulatory program, such as genetic or chemical perturbations, alter the resulting transcriptome (Dixit et al., 2016; Adamson et al., 2016). If reliable, virtual cells stand to enable a new frontier of in-silico drug discovery.

The central object of observed gene expression is the count matrix: a sparse, zero-inflated, high-dimensional integer-valued tensor recording how many mRNA molecules of each gene were captured from each cell. These properties make direct generative modeling on raw counts remarkably difficult for most models. Autoregressive factorizations impose a sequential ordering on genes, an inductive bias misaligned with their natural exchangeability. Flow matching tends to collapse to the mean under the sparsity of count data. Masked diffusion is the only family that recovers reasonable structure on raw counts (Zhang et al., 2026; Bhattacharya et al., 2026).

These difficulties on raw counts motivate a latent-space approach. Autoregressive models have demonstrated predictable scaling laws on language data (Hoffmann et al., 2022), an appealing property as virtual cell models scale up, and they operate naturally over discrete tokens. This motivates a discrete latent representation.

We introduce `ExpressionVAE`, a scalar-quantized variational autoencoder that encodes each cell as a fixed-length sequence of discrete codes, paired with a perturbation-conditioned prior over those codes. Scalar quantization

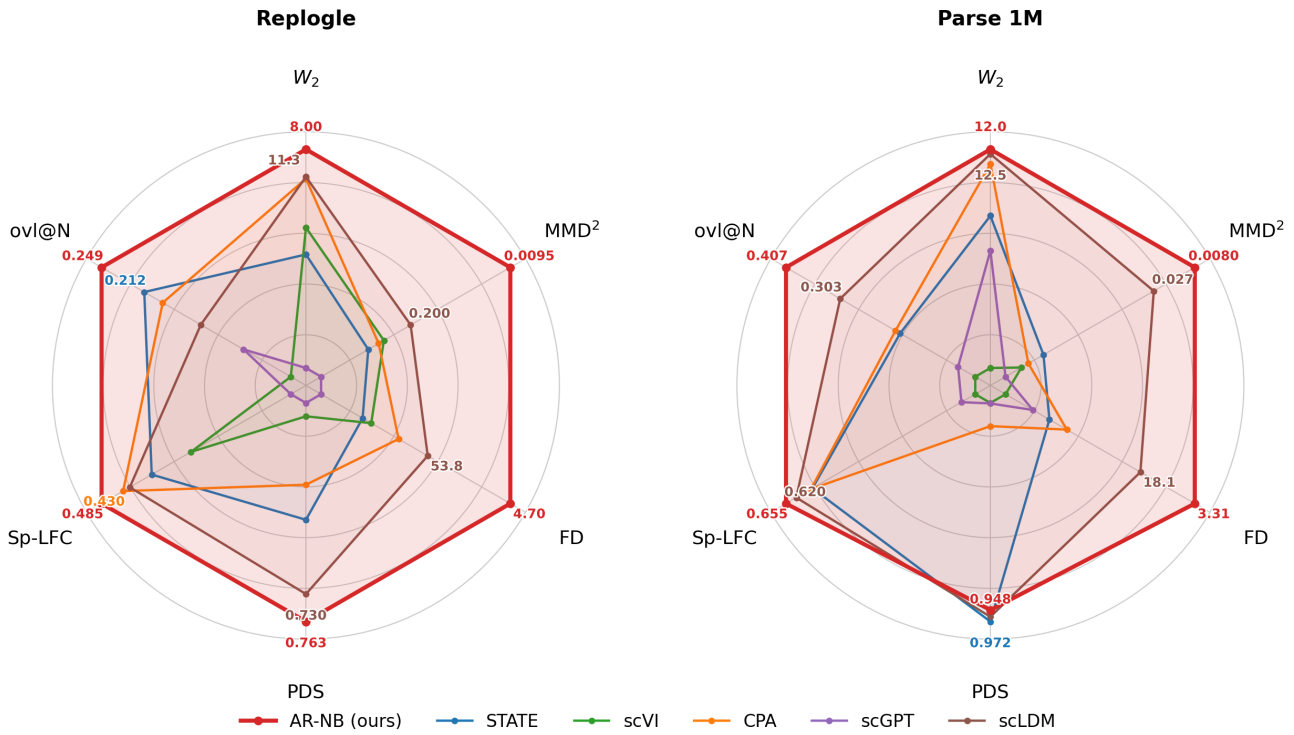

Figure 1. **ExpressionVAE on the outer envelope of both metric families.** Six-axis comparison of our representative model (Autoregressive prior with Negative Binomial head) against five baselines on Replogle (left) and Parse 1M (right), spanning three distribution metrics (W_2, MMD^2, FD) and three cell-eval metrics (PDS, Sp-LFC, ovl@N). ExpressionVAE model holds the outermost ring on all six axes on Replogle and five of six on Parse 1M (STATE narrowly leads PDS). It is the only method on the frontier of both metric families: baselines strong on the distribution axes retreat on the ranking axes, and those strong on ranking collapse on the distribution axes. Each axis is rescale and adjusted so that better always points outward.

commits the encoder to a finite codebook of cell-state representations that any standard discrete prior can model exactly. To control for the choice of prior, we train two from different generative-model families, autoregressive (AR) and masked discrete diffusion (MDLM); any gain that survives the swap is attributable to the latent representation rather than to a particular prior architecture.

Beyond the latent and prior, two further design questions recur in the single-cell perturbation literature, and we treat each as a controlled experiment. First, single-cell models vary substantially in their decoder output head, and published rankings across cross-entropy, MSE, hurdle, and negative-binomial heads are inconsistent in ways that often track the choice of evaluation metric rather than any property of the heads themselves. We run a within-architecture head ablation on the same trained backbone to expose the underlying axis. Second, we evaluate the frozen encoder on a CRISPRi reversion benchmark to test whether the resulting design choices reflect a generalizable inductive bias rather than over-fitting our chosen metric suite.

Concretely, our contributions are:

- **Discrete-latent perturbation modeling.** We introduce a discrete-latent representation learning approach for single-cell perturbation modeling, and show that pairing `ExpressionVAE` with either an autoregressive or a masked-diffusion prior achieves state-of-the-art on the distributional metrics and leads on most cell-eval state metrics on both Replogle and Parse 1M, with MMD^2 and Fréchet distance roughly 3–20$\times$ lower than the strongest continuous-latent baseline. The two priors achieve essentially identical performance, isolating the gain to the discrete latent representation rather than to the prior architecture (Section 4.1).

- **A measurement-theoretic account of decoder-head choice.** We show that the apparent conflicts between published evaluation metrics resolve into a clean two-group structure once a single design axis, the richness of the inference-time sampling distribution, is made explicit. Every metric is either *variance-sensitive*, rewarding heads that reproduce within-perturbation spread, or *mean-sensitive*, rewarding heads that emit clean per-gene means (Section 4.2).

- **Usefulness of the latent space.** On a held-out

CRISPRi reversion benchmark of 1,732 perturbations under inflammatory cytokine stress (Wong et al., 2026), the hurdle decoder identified by our controlled ablation is the strongest pairing on biological selectivity, and the frozen encoder effectively matches a $10\times$-larger foundation model (scGPT) on enrichment AUC for perturbation-ranking. This shows that the discrete-latent perturbation prior captures inflammation-axis structure effectively (Section 4.3).

2. Related Work

We draw from two literatures: single-cell generative and perturbation-response models, and the latent modeling toolkit from vision and language.

Generative and perturbation-response models. scVI (Lopez et al., 2018a;b) established the canonical Gaussian-latent, Negative Binomial-likelihood VAE recipe, elaborated by the conditional-VAE thread of scGen, trVAE, and CPA (Lotfollahi et al., 2019; 2020; 2023) and the latent-diffusion thread of scDiffusion (Luo et al., 2024), CFGen (Palma et al., 2024), and scLDM (Palla et al., 2025) with flow-matching priors (Lipman et al., 2022; Peebles and Xie, 2023); the Hurdle head we evaluate traces to MAST (Finak et al., 2015). Beyond this lineage, perturbation-response models pretrained on tens to hundreds of millions of cells (Geneformer (Theodoris et al., 2023), scGPT (Cui et al., 2024), scFoundation (Hao et al., 2024), STATE (Adduri et al., 2025), Tahoe-x1 (Gandhi et al., 2025)) increasingly condition on ESM-2 embeddings (Lin et al., 2023). All embed raw counts into continuous vectors. A concurrent line generates directly in the discrete count domain, via Score Entropy Discrete Diffusion (DCM (Bhattacharya et al., 2026; Lou et al., 2023)), absorbing-state masked diffusion in the D3PM (Austin et al., 2021) lineage (Lingshu-Cell (Zhang et al., 2026), X-Cell (Wang et al., 2026a), scDiVa (Wang et al., 2026b)), or Stack's in-context approach (Dong et al., 2026). We evaluate on Replogle Perturb-seq (Replogle et al., 2022) and the Parse cytokine atlas (Oesinghaus et al., 2025), with phenotypic reversion (Wong et al., 2026) as our NFκB ranking experiment.

Latent generative modeling: tokenizers and priors. On the tokenizer side, the community relies heavily on VQ-VAE as the go-to method for obtaining discrete tokens. VQ-VAE (van den Oord et al., 2017; Razavi et al., 2019) and MaskGIT (Chang et al., 2022) introduced learnable discrete codes and masked parallel decoding, Latent Diffusion (Rombach et al., 2022) established the continuous-latent plus diffusion-prior recipe, and FSQ (Mentzer et al., 2023) eliminated the codebook collapse documented in VQ literature including in single cell domain (Liu et al., 2025). On the prior side, the absorbing-state masked discrete diffusion family has matured rapidly from D3PM (Austin et al., 2021) through SEDD (Lou et al., 2023), MDLM (Sahoo et al., 2024), and MD4 (Shi et al., 2024) to RADD (Ou et al., 2025), which unifies it with any-order autoregression. We combine learned discrete latents with flexible generative priors, a combination no existing model explores.

3. Method

We propose a two-stage pipeline (Fig. 2). Stage 1 trains an `ExpressionVAE` that compresses each cell's gene-expression vector into a fixed-length sequence of discrete codes via a finite scalar quantization (FSQ) bottleneck (Mentzer et al., 2023). Stage 2 freezes the VAE and trains a perturbation-conditioned generative prior over the code sequence, conditioned on an ESM2-3B protein-language-model embedding of the targeted gene and a learned additive cell-type signal. At inference, the prior is queried with the embedding of an unseen perturbation and the resulting latent is decoded into a predicted gene-expression profile. The two design axes varied across the experiments are the VAE *output head* (Section 4.2) and the choice of *prior* (Section 4.1).

3.1. Stage 1: ExpressionVAE

Encoder and decoder. The encoder maps a count vector to a set of L latent tokens via a cross-attention transformer. Each gene g is first embedded as $\mathbf{e}_g = f_{\text{in}}(x_g, \mathbf{v}_g)$, where f_{in} is an MLP applied to the log1p-normalized scalar count x_g and modulated by a learned gene-identity token $\mathbf{v}_g \in \mathbb{R}^d$. L learnable latent query tokens cross-attend over the G gene embeddings through a stack of transformer layers, producing a per-cell latent matrix $\mathbf{Z} \in \mathbb{R}^{L\times d}$. A symmetric cross-attention transformer decodes the quantized codes back to gene space: G learnable gene query tokens cross-attend over the projected code embeddings, yielding a per-gene representation $\mathbf{H} \in \mathbb{R}^{G\times d}$.

FSQ bottleneck. The encoder output $\mathbf{Z}$ is quantized per-dimension by FSQ (Mentzer et al., 2023), which replaces the learned Vector Quantized (VQ) codebook with a parameter-free per-dimension rounding scheme. Given a level vector $\boldsymbol{\ell} = (\ell_1, \ldots, \ell_d) \in \mathbb{Z}^d_{>0}$, each dimension is bounded via $\tanh$ and rounded to the nearest integer on a uniform grid:

$$\hat{z}_k = \text{round}\left(\frac{\ell_k - 1}{2} \cdot \tanh(z_k)\right), \qquad k = 1, \ldots, d.$$

The result is a per-cell sequence of L discrete codes $\mathbf{c} = (c_1, \ldots, c_L)$ drawn from a vocabulary of size $\prod_k \ell_k$, eliminating the codebook collapse typically seen in VQ, also observed in single-cell literature (Liu et al., 2025). We use $L = 64$ tokens with codebook size $K = 512$. A Gaussian-latent variant used in the Appendix also uses $L = 64$ latent tokens with $\beta_{\text{KL}} = 2 \times 10^{-6}$.

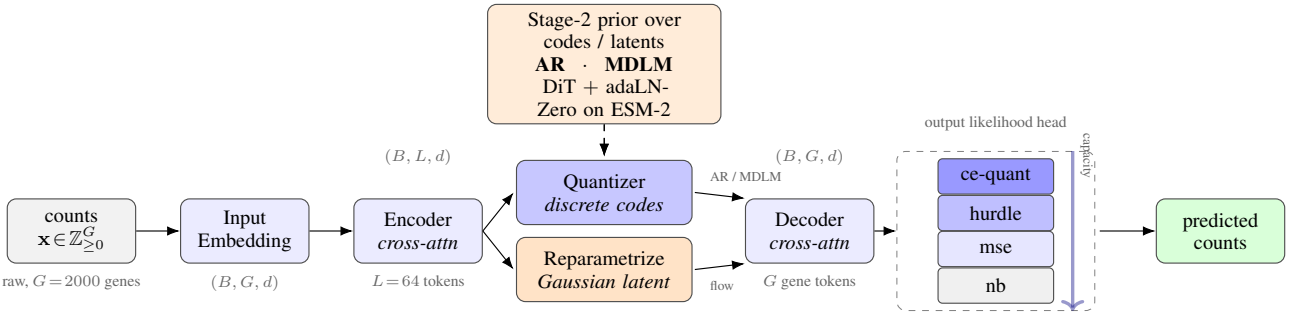

Figure 2. A two-stage pipeline. Stage 1 (the row of boxes): a cross-attention transformer encoder maps raw count vectors $\mathbf{x} \in \mathbb{Z}^G_{\geq 0}$ to L=64 latent tokens, which are either FSQ quantised into discrete codes (consumed by AR or MDLM priors) or reparametrised as continuous Gaussian latents (consumed by flow). A symmetric cross-attention decoder maps these back to per-gene representations, which one of four output likelihood heads converts to predictions. Stage 2 (sidebar): a DiT prior over the frozen codes / latents, conditioned on the ESM-2 perturbation embedding and optionally cell-type embedding via adaLN-Zero.

Output heads. An output head maps the per-gene representation $\mathbf{H}$ to a conditional distribution $p_\theta(\mathbf{x} \mid \mathbf{H})$. We evaluate four head families, corresponding to standard choices in the single-cell modeling literature, and refer to them by the labels used in Section 4.2.

`ce-quant` places a categorical over $[0, c_{\max}]$ per gene with logits produced by a linear projection of $\mathbf{h}_g$, trained against quantile-binned integer targets and sampled at inference (Section 3.1 below).

`hurdle` (Finak et al., 2015) parameterises a Bernoulli zero-gate $z_g \sim \mathrm{Bern}(1 - p_{0,g})$ together with a regressed positive magnitude μ_g; the loss combines BCE on the gate with MSE on non-zero entries, and inference samples the gate while emitting μ_g deterministically.

`mse` regresses log1p-normalised counts directly, with $p(x_g \mid \mathbf{h}_g) = \mathcal{N}(\mu_g, 1)$, and emits μ_g with no stochastic step.

The negative binomial (NB) family parameterises $\mathrm{NB}(\mu_g, \theta_g)$ per gene with library-size factorisation, following (Lopez et al., 2018a; Palla et al., 2025): $\mu_g = \ell_n \cdot \rho_g$, where $\boldsymbol{\rho} = \mathrm{softmax}_g(\mathbf{w}^\top \mathbf{H})$ is a probability simplex over genes, $\ell_n = \sum_g x_g$ is the per-cell library size, and θ_g is a learned cell-independent dispersion. This family admits two distinct `predict()` rules at inference time without changing trained weights: `nb-sample` draws from $\mathrm{NB}(\mu, \theta)$, and `nb-deter` returns μ directly. We treat these as two separate evaluation points throughout, since the choice of inference rule materially shifts which evaluation metrics the head wins (Section 4.2).

Quantile binning for the categorical head. The `ce-quant` head requires integer targets, so we quantile-bin the raw counts into a vocabulary of 20 bins chosen from the empirical training count distribution: 8 singleton bins for counts 0–7 capture the dominant low-count regime exactly, 7 bins covering 8–63 handle moderate counts at progressively coarser resolution, and 5 bins (64–95, 96–127, 128–191, 192–255, ≥ 256) cover the high-expression tail. Tokenisation maps each raw count to its bin index; detokenisation maps each predicted bin back to the empirical conditional mean of counts within that bin range, not the arithmetic midpoint, since the within-bin distribution is heavy-tailed and the bias matters most for the unbounded ≥ 256 bucket. This scheme captures 100% of top-DEG per-perturbation means on Parse and 99.69% on Replogle within bounded bins.

3.2. Stage 2: Discrete prior over latent codes

Stage 2 freezes the VAE and trains a generative prior over the code sequence $\mathbf{c} = (c_1, \ldots, c_L)$, conditioned on a projected ESM2-3B perturbation embedding $\mathbf{p}$ and an additive cell-type signal. Both prior families share a DiT backbone with adaLN-Zero conditioning on $\mathbf{p}$ and the cell-type vector. The autoregressive (AR) prior factorises as $p(\mathbf{c} \mid \mathbf{p}) = \prod_{l=1}^{L} p(c_l \mid c_{<l}, \mathbf{p})$ via a causally-masked DiT, requiring L sequential decoding steps at inference. MDLM (Sahoo et al., 2024) trains a bidirectional DiT denoiser under an absorbing-state masking schedule $\alpha_t = 1 - t$, with the standard masked-token loss

$$\mathcal{L}_{\text{MDLM}} = \mathbb{E}_{t, \mathbf{z}_t}\left[\frac{1}{t} \sum_{i:\, z_t^i = \texttt{[M]}} -\log p_\theta(c^i \mid \mathbf{z}_t, \mathbf{p})\right],$$

and denoises all L tokens in a single forward pass. We use the same backbone, parameter count, and training schedule for both, so any gap in Section 4.1 reflects the prior family rather than capacity.

3.3. Inference

Given an unseen perturbation, the corresponding ESM2-3B embedding $\mathbf{p}^*$ is passed to the trained prior, which samples a latent sequence $\hat{\mathbf{c}} \sim p_\theta(\cdot \mid \mathbf{p}^*)$ via L autoregressive steps for AR or D denoising steps for MDLM. The frozen VAE decoder maps $\hat{\mathbf{c}}$ back to gene space, and the chosen

output head emits a predicted profile $\hat{\mathbf{x}} \in \mathbb{R}^G$ following its `predict()` rule. The same frozen encoder is reused without any fine-tuning for the reversion ranking evaluation in Section 4.3.

4. Experiments

Datasets. We evaluate on two large-scale single-cell perturbation screens. The train and heldout splits are matched to scLDM (Palla et al., 2025). Parse 1M consists of cytokine perturbations across approximately one million cells, with a CD4 Naive holdout of 27 cytokines. The Replogle essential-genome CRISPRi screen (Replogle et al., 2022) provides a HepG2 holdout of 372 knock-outs.

Metrics. We evaluate at two complementary levels. *Distribution* metrics (W_2, MMD^2 with an RBF kernel, and Fréchet distance FD), following (Palla et al., 2025), compare the full predicted cell population against the true cells in PCA space. The *cell-eval* per-perturbation metrics from (Adduri et al., 2025) score the predicted transcriptional change relative to control: *PR-AUC* and *Spearman sig* are built on FDR-based DE-test calls, PDS measures perturbation specificity from the pseudobulk effect vector, *Spearman LFC* and *overlap@N* rank genes by predicted $\log_2$ fold change, and *Pearson* Δ correlates the full per-gene Δ vector. All numbers report mean $\pm$ SE across four seeds, with full definitions in Appendix C.

4.1. Discrete latents outperform continuous baselines

Existing single-cell perturbation models build on continuous latent representations (scVI, CPA, scLDM) or operate directly in expression space (scGPT, STATE). Discrete-latent representation learning, despite being the foundation of the strongest generative models in language and vision, has not previously been applied to perturbation modeling. We close this gap. The `ExpressionVAE` encodes each cell as a fixed-length sequence of discrete codes drawn from a scalar-quantized bottleneck, and a perturbation-conditioned prior, either autoregressive (AR) or masked discrete diffusion (MDLM), models the joint distribution over those codes. We compare against six baselines spanning continuous latent VAEs, conditional perturbation models, expression-space foundation models, and a non-parametric mean baseline: scLDM (Palla et al., 2025), scVI (Lopez et al., 2018a), CPA (Lotfollahi et al., 2023), scGPT (Cui et al., 2024), STATE (Adduri et al., 2025), and PerturbMean. The results of Replogle and Parse 1M are reported in Table 1.

For each prior we report two rows, corresponding to the NB-sample and MSE output heads. These sit at the sampling and deterministic ends of the head spectrum analyzed in Section 4.2 and trade off across metric families; the practical takeaway here is that our model achieves the strongest result in each family. Three observations stand out.

3–20× gains on distributional metrics. The headline result is on W_2, MMD^2, and FD, where the discrete-latent model improves over the strongest continuous-latent baseline (scLDM) by $\sim$10–20$\times$ on Replogle and $\sim$3–6$\times$ on Parse 1M, on MMD^2 and FD. On Replogle, MMD^2 drops from 0.200 to 0.009 and FD from 53.75 to 4.70; on Parse 1M, MMD^2 drops from 0.027 to 0.008 and FD from 18.14 to 3.31. Expression-space methods (scGPT, STATE) and continuous latent baselines (scVI, CPA) lag the discrete-latent model by one to three orders of magnitude on these metrics.

Two priors, same numbers. AR and MDLM, drawn from different generative-model families and commonly viewed as alternatives, achieve effectively identical performance across all metrics on both datasets. This is direct evidence that the gain is from the discrete-latent representation rather than from a particular prior architecture, since the two priors share only the encoder and the codebook.

Top of both metric families. The model achieves the strongest result on every distributional metric and most DE-discovery metrics *and* on every mean-ranking metric, on both datasets. No baseline reaches the top of both families. Existing methods are competitive on distributional metrics or on ranking metrics, but never both: scLDM is the strongest baseline on distributional metrics yet trails on *Sp-LFC* and *overlap@N*, while PerturbMean and STATE are competitive on PDS and mean-ranking metrics but collapse on W_2, MMD^2, and FD. The two-rows-per-prior structure that produces this pattern is not incidental; Section 4.2 shows it reflects a single underlying axis governed by inference-time stochasticity.

4.2. Sampling-richness axis explains decoder-head behavior

A practical question for any single-cell perturbation model is which decoder head to use: cross-entropy, hurdle, MSE, negative binomial. The literature offers no consistent answer. Different papers report different metrics, and different metrics rank these heads differently. The two rows per prior in Table 1 make the conflict visible within a single backbone: the same trained model produces noticeably different metric profiles depending only on the head. We show here that this is not noise. The standard evaluation metrics partition into two groups, and within each group the head ordering is monotonic in a single underlying axis, the richness of the inference-time sampling distribution, with the two groups responding in opposite directions along it.

Table 1. **Discrete-latent perturbation modeling against six baselines on Replogle and Parse 1M.** For each prior (AR, MDLM) we report two rows, corresponding to the NB-sample and MSE output heads (Section 4.2). Bold = best in column for that dataset, $*$ = scores are re-used from original reports.

Dataset	Prior	$W_2 \downarrow$	$\text{MMD}^2 \downarrow$	FD $\downarrow$	PDS $\uparrow$	PR-AUC* $\uparrow$	Sp-sig* $\uparrow$	Sp-LFC $\uparrow$	ovl@N $\uparrow$	P-Δ^* $\uparrow$
Replogle	AR	**8.001**	**0.009**	**4.70**	**0.763**	0.295	0.789	0.480	0.170	0.307
		9.032	0.099	35.06	0.716	0.230	0.793	0.508	**0.255**	0.381
	MDLM	8.121	0.013	5.29	0.756	**0.297**	**0.799**	0.495	0.192	0.318
		9.091	0.105	35.70	0.698	0.219	0.793	**0.540**	0.248	0.362
	STATE	20.580	0.730	366.64	0.641	0.210	0.721	0.357	0.212	**0.420**
	ScVI	17.359	0.453	284.47	0.517	0.190	0.598	0.257	0.085	0.350
	CPA	11.510	0.532	126.81	0.599	0.190	0.370	0.430	0.196	0.380
	ScGPT	34.166	3.087	1247.68	0.501	0.100	0.550	0.002	0.126	0.385
	ScLDM ($\omega = 1$)	11.292	0.200	53.75	0.730	–	–	0.413	0.163	–
	PerturbMean	–	–	–	0.616	0.190	0.190	0.246	0.162	–
Parse 1M	AR	**11.952**	**0.008**	**3.31**	0.948	**0.498**	0.949	**0.655**	**0.407**	0.762
		22.080	0.646	375.41	0.632	0.157	0.926	0.618	0.140	0.747
	MDLM	11.969	0.008	3.50	0.956	0.485	**0.960**	0.639	0.401	**0.763**
		22.152	0.654	378.10	0.624	0.141	0.901	0.646	0.144	0.733
	STATE	19.111	0.714	312.34	0.972	–	–	0.565	0.188	–
	ScVI	35.508	1.372	1233.11	0.519	–	–	0.010	0.044	–
	CPA	13.534	1.117	181.32	0.567	–	–	0.570	0.197	–
	ScGPT	22.870	2.203	523.93	0.520	–	–	0.057	0.077	–
	ScLDM ($\omega = 1$)	12.457	0.027	18.14	0.962	–	–	0.620	0.303	–
	PerturbMean	–	–	–	**0.973**	–	–	0.467	0.118	–

Five heads on a sampling-richness gradient. We evaluate five heads on the same trained `ExpressionVAE` backbone. `ce-quant` parameterizes a categorical over $[0, c_{\max}]$ per gene and samples from the softmax. `nb-sample` parameterizes a negative binomial $\text{NB}(\mu, \theta)$ per gene and emits a draw from it. `hurdle` parameterizes a Bernoulli zero-gate combined with a deterministic positive magnitude μ, sampling only the gate. `mse` emits a deterministic real-valued μ everywhere. `nb-deter` is just an NB head without sampling, it deterministically returns μ. Read in the order `ce-quant` $\rightarrow$ `nb-sample` $\rightarrow$ `hurdle` $\rightarrow$ `mse` $\rightarrow$ `nb-deter`, the heads form a gradient in the richness of the inference-time predictive distribution: full count sampling, parametric count sampling, sampled zero-gate over deterministic magnitude, fully deterministic, and fully deterministic plus structurally anti-sparse (due to softmax over genes in NB).

Two groups, opposing trends. Figure 3 reports all five heads under both priors as per-metric heatmaps, with columns ordered so that the variance-sensitive group precedes the mean-sensitive group. The structure is immediate from the color gradient. The variance-sensitive columns (all three distribution metrics plus DE-discovery: *disc_l1*, *PR-AUC*, *Spearman sig*) darken toward the sampling end of the head ordering, the top of each prior block; the mean-sensitive columns (*Spearman LFC*, *overlap@N*, *Pearson* Δ) darken toward the deterministic end, the bottom. The variance-sensitive collapse is most dramatic on Parse 1M, where moving the head from sampling to deterministic shifts FD by more than two orders of magnitude (e.g., $1.84 \rightarrow 454.1$ along the AR block); on Replogle the same trend holds in direction but is compressed in magnitude, consistent with that dataset's lower count dispersion.

The mechanism follows from how each metric is computed (Appendix C). The prior draws a fresh latent per cell (Section 3.3), so the predicted population is already spread across cell states for every head; what the head controls is the conditional count noise at a fixed latent. A sampling head adds the per-gene over-dispersion and zero-inflation of real counts on top of each latent's mean, whereas a deterministic head returns the mean, leaving an *under-dispersed* population. The variance-sensitive metrics penalise this. Fréchet distance, MMD^2, and W_2 rise as the predicted spread falls short of the true cells; *PR-AUC* and *Spearman sig* compute per-gene FDRs from a Mann–Whitney U test that, with too little predicted-side spread it saturates almost any significant mean shift and stops ranking genes by perturbation strength; and PDS sums an L_1 over the full per-gene effect vector, where the under-dispersed, zero-poor pseudobulk is biased across the many low-count genes. The mean-sensitive metrics instead concentrate on the strongly-perturbed genes, where the deterministic mean is cleanest and count noise only adds finite-sample variance: *Spearman LFC* and *overlap@N* are rank-based on the real-DE set, and

	W2 ↓	MMD² ↓	FD ↓	Disc-L1 ↑	PR-AUC ↑	Spearman sig ↑	Spearman L2FC ↑	DEG Overlap ↑	Pearson Δ ↑
AR ce-quant	7.864 ±0.010	0.009 ±0.0003	4.540 ±0.080	0.740 ±0.0040	0.282 ±0.0010	0.793 ±0.0050	0.513 ±0.012	0.164 ±0.0030	0.310
AR nb-sample	8.001 ±0.010	0.009 ±0.0003	4.700 ±0.090	0.763 ±0.0030	0.295 ±0.0010	0.789 ±0.0040	0.480 ±0.0090	0.170 ±0.0020	0.307
AR hurdle	7.900 ±0.010	0.013 ±0.0004	5.470 ±0.110	0.745 ±0.0020	0.298 ±0.0010	0.784 ±0.0030	0.461 ±0.0060	0.213 ±0.0010	0.308
AR mse	9.032 ±0.017	0.099 ±0.0007	35.06 ±0.240	0.716 ±0.0030	0.230 ±0.0020	0.793 ±0.0040	0.508 ±0.0090	0.255 ±0.0020	0.381
AR nb (deter)	11.94 ±0.013	0.303 ±0.0014	100.2 ±0.380	0.742 ±0.0010	0.220 ±0.0010	0.778 ±0.0050	0.485 ±0.0090	0.249 ±0.0020	0.378
MDLM ce-quant	8.051 ±0.020	0.013 ±0.0004	5.360 ±0.150	0.715 ±0.0030	0.275 ±0.0020	0.774 ±0.0050	0.439 ±0.0040	0.158 ±0.0020	0.297
MDLM nb-sample	8.121 ±0.014	0.013 ±0.0004	5.290 ±0.100	0.756 ±0.0020	0.297 ±0.0020	0.799 ±0.0050	0.495 ±0.0070	0.192 ±0.0020	0.318
MDLM hurdle	8.042 ±0.0080	0.017 ±0.0005	6.380 ±0.070	0.728 ±0.0010	0.290 ±0.0010	0.783 ±0.0040	0.490 ±0.0090	0.216 ±0.0010	0.308
MDLM mse	9.091 ±0.021	0.105 ±0.0011	35.70 ±0.380	0.698 ±0.0040	0.219 ±0.0010	0.793 ±0.0070	0.540 ±0.0070	0.248 ±0.0010	0.362
MDLM nb (deter)	12.09 ±0.010	0.316 ±0.0008	102.4 ±0.200	0.710 ±0.0050	0.214 ±0.0010	0.764 ±0.0030	0.519 ±0.0080	0.244 ±0.0020	0.359 ±0.0010

	W2 ↓	MMD² ↓	FD ↓	Disc-L1 ↑	PR-AUC ↑	Spearman sig ↑	Spearman L2FC ↑	DEG Overlap ↑	Pearson Δ ↑
AR ce-quant	11.99 ±0.011	0.004 ±0.0001	1.860 ±0.020	0.967 ±0.0030	0.506 ±0.0020	0.948 ±0.0070	0.749 ±0.018	0.413 ±0.0050	0.720
AR nb-sample	11.96 ±0.0090	0.008 ±0.0002	3.150 ±0.050	0.947 ±0.0030	0.486 ±0.0040	0.963 ±0.0050	0.827 ±0.020	0.402 ±0.0050	0.764
AR hurdle	12.07 ±0.0070	0.012 ±0.0003	5.300 ±0.110	0.912 ±0.0020	0.419 ±0.0050	0.940 ±0.0030	0.745 ±0.017	0.294 ±0.0040	0.725
AR mse	22.11 ±0.108	0.646 ±0.0047	376.8 ±4.56	0.635 ±0.0040	0.146 ±0.0020	0.912 ±0.0070	0.789 ±0.0040	0.474 ±0.017	0.749
AR nb (deter)	23.11 ±0.038	0.855 ±0.0017	455.7 ±1.75	0.621 ±0.0050	0.123 ±0.0020	0.784 ±0.0060	0.740 ±0.0050	0.457 ±0.0070	0.779
MDLM ce-quant	12.00 ±0.0040	0.005 ±0.0001	2.220 ±0.040	0.967 ±0.0020	0.509 ±0.0020	0.948 ±0.0070	0.724 ±0.0090	0.419 ±0.0050	0.710
MDLM nb-sample	11.97 ±0.0070	0.008 ±0.0001	3.400 ±0.060	0.953 ±0.0030	0.495 ±0.0040	0.964 ±0.0060	0.814 ±0.022	0.416 ±0.0050	0.768
MDLM hurdle	12.08 ±0.010	0.013 ±0.0002	5.540 ±0.080	0.896 ±0.0040	0.419 ±0.0050	0.932 ±0.0050	0.751 ±0.030	0.310 ±0.0040	0.717
MDLM mse	22.07 ±0.073	0.646 ±0.0031	374.3 ±2.78	0.628 ±0.0030	0.148 ±0.0030	0.927 ±0.0060	0.797 ±0.010	0.466 ±0.016	0.737
MDLM nb (deter)	22.98 ±0.047	0.848 ±0.0023	448.6 ±2.07	0.616 ±0.0050	0.115 ±0.0010	0.759 ±0.012	0.765 ±0.0070	0.457 ±0.0060	0.769 ±0.0010

Figure 3. **Output-head ablation on Replogle (top) and Parse 1M (bottom).** Rows are five output heads under AR and MDLM priors on the same trained backbone; columns are grouped into the two functional families, variance-sensitive (distribution and DE-discovery) and mean-sensitive (mean-ranking). Cells shaded per column, darker indicates better; mean over four seeds, SE in $\pm$ notation. Parse 1M DE-based metrics in this figure use a DE-significance threshold of $\epsilon = 0.5$.

Pearson Δ correlates the full Δ vector but, as a correlation, is dominated by the large-magnitude genes. PDS and *Pearson* Δ make the split concrete: both read the full per-gene vector, but L_1 weights the sparse bulk (variance-sensitive) while correlation weights the peaks (mean-sensitive).

The NB pair isolates the sampling axis. A natural confound is that `ce-quant` differs from `mse` on multiple axes simultaneously: discreteness, parametric form, output dimension, and training loss. To isolate sampling versus deterministic prediction from these confounds, we exploit that `nb-sample` and `nb-deter` share trained weights and differ only in `predict()`. Switching from μ to a draw from $\mathrm{NB}(\mu, \theta)$ moves NB from the bottom tier to the top tier of the variance-sensitive group (e.g., on Parse, $\mathrm{FD} : 454.06 \rightarrow 3.31$ and $\mathrm{PR\text{-}AUC} : 0.129 \rightarrow 0.498$ under AR), and reverses the direction on the mean-sensitive group (e.g., on Replogle, $\mathrm{overlap@N} : 0.281 \rightarrow 0.170$ under AR). The shift is largest on the distribution metrics and milder elsewhere, but the direction is consistent within each group. Because no other component of the model changes, the entire effect is localized to the sampling step at inference. This rules out the confounds and pins the cross-group structure on the sampling axis.

What metric measures what, and a recommendation. The variance-sensitive group rewards recovery of within-perturbation cell variance, expressed once directly at the cell-distribution level (the distribution metrics) and once filtered through DE testing (the DE-discovery metrics). The mean-sensitive group rewards clean per-gene mean estimates and is, by construction, penalised by sampling noise. The practical implication is that head choice is governed by what the practitioner intends to measure. We recommend reporting at least one metric from each group, and selecting the head to match the downstream task: a sampling head (`ce-quant` or sampled NB) for use cases that consume predicted cell distributions (synthetic-cohort DE, virtual screens, power calculations), and a deterministic-mean head (`mse` or NB-mean) for gene-list ranking against a reference. Reporting any single metric in isolation, or comparing models that report metrics from different groups, is misleading.

4.3. Usefulness of latent space for phenotypic reversion

Sections 4.1 and 4.2 establish that a discrete-latent perturbation model with the right decoder head dominates within-distribution metrics, and that head choice is governed by an inference-time sampling axis with a clean metric-to-head map. To test whether these design choices reflect a generalizable inductive bias rather than over-fitting the chosen metric set, we evaluate the *frozen* encoder on a biological discovery task. No information about the biological task's dataset enters training; the encoder is used only to embed cells, and metrics are computed directly on the resulting latents.

Benchmark. The TeloHAEC phenotypic-reversion benchmark (Wong et al., 2026) is a CRISPRi screen of 1,732 perturbations across 864,115 cells under IL-1β and TNF-α inflammation. The task is to rank perturbations capable of reverting the inflammatory state, with the canonical NF-κB activators (TNFRSF1A, TRADD, JUNB, JUND, NFKB1, NFKB2) as positive controls. We report three metrics on the frozen latents: per-perturbation Calinski-Harabasz scores among treated cells (Gene CH (T)) and untreated cells (Gene CH (U)), measuring latent separability; and Enrichment AUC, the rank-AUC of the NF-κB positive controls within the cosine-distance reversion ordering of the 1,732 perturbation centroids (random baseline 0.46).

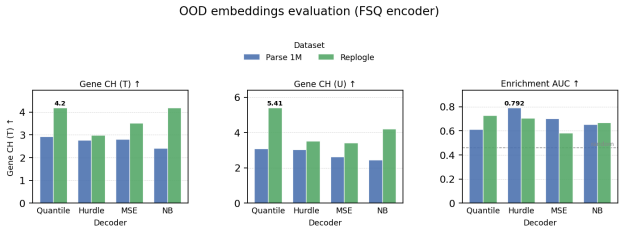

Figure 4. **Latent space evaluation on the TeloHAEC phenotypic-reversion benchmark.** Frozen `ExpressionVAE` encoders trained on Replogle and Parse 1M. Dashed line on the right shows the random baseline (0.46); annotated values mark the best configuration per panel.

The decoder transfers; the encoder pairing is dataset-dependent. The Hurdle decoder identified by the controlled ablations is the strongest pairing on Enrichment AUC on Parse 1M dataset (0.792); the decoder choice transfers cleanly without fine-tuning. Both axes contribute measurably to performance, confirming that the decoder is not just a likelihood-modeling decision but shapes the representation itself.

Quantiled inputs dominate separability. On the per-perturbation Calinski-Harabasz scores, FSQ + Quantile is the strongest configuration on both Gene CH (T) and Gene CH (U) (4.20, 5.41 on Replogle). Hard binning in inputs commits the encoder to a finite codebook of phenotypic states, producing sharper per-perturbation cluster structure in the latent space. This is consistent with the distributional-metric gap reported in Section 4.1. Overall, discrete latents allocate representational capacity in a way that preserves cell-state distinctions both within and outside the training distribution.

Comparison with scGPT. On the same benchmark, scGPT (Cui et al., 2024) achieves Enrichment AUC $= 0.79$. Our strongest Parse-trained configuration reaches 0.79, within typical evaluation noise of scGPT despite scGPT consuming roughly $10\times$ more pre-training data. The point is not a SOTA claim on the TeloHAEC benchmark, but a representation-efficiency claim: a discrete-latent perturba-

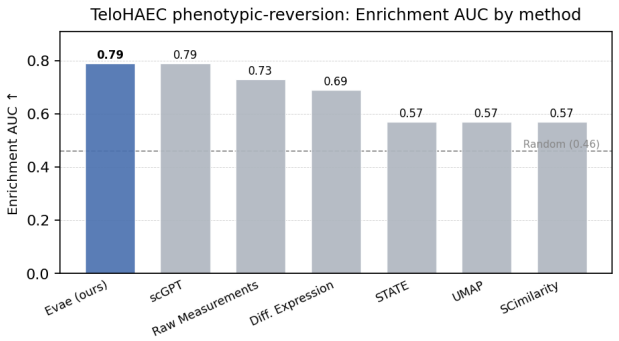

Figure 5. **Enrichment AUC for the TeloHAEC phenotypic-reversion benchmark.** Dashed line on the right shows the random baseline (0.46); annotated values mark the best configuration per panel.

tion prior trained on a single 1M-cell dataset effectively matches a foundation model trained on tens of millions of cells, on a biologically meaningful ranking task. The explicit zero-gate decoder identified by our ablation transfers cleanly off the training distribution. We acknowledge that this benchmark has several distribution shifts such as cell type, perturbation set, biological context, etc, making it hard to identify which axis of generalization the model actually uses with the limited training dataset.

5. Conclusion

We introduced `ExpressionVAE`, a discrete-latent perturbation model for single-cell data, and showed that pairing a scalar-quantized bottleneck with either an autoregressive or a masked-diffusion prior achieves state-of-the-art on distributional metrics and leads on most cell-eval metrics on Replogle and Parse 1M. Both priors achieve effectively identical performance, isolating the gain to the discrete latent rather than to a particular prior. A controlled output-head ablation further showed that decoder-head choice in this setting is governed by a single inference-time design axis, sampling versus deterministic prediction, with standard evaluation metrics partitioning into two groups along it, variance-sensitive and mean-sensitive; this resolves the apparent conflicts between published metric rankings into a clean two-way mapping between heads and downstream tasks. The hurdle decoder identified by the ablation transfers as the strongest selectivity pairing on a held-out CRISPRi reversion benchmark, where the frozen encoder effectively matches a $10\times$ larger foundation model at a fraction of the training data. Taken together, these results suggest that a compact discrete latent captures perturbation-axis structure effectively and is a promising step toward virtual cells for in-silico discovery.

References

B. Adamson, T. M. Norman, M. Jost, M. Y. Cho, J. K. Nuñez, Y. Chen, J. E. Villalta, L. A. Gilbert, M. A. Horlbeck, M. Y. Hein, et al. A multiplexed single-cell crispr screening platform enables systematic dissection of the unfolded protein response. *Cell*, 167(7):1867–1882, 2016.

A. K. Adduri, D. Gautam, B. Bevilacqua, A. Imran, R. Shah, M. Naghipourfar, N. Teyssier, R. Ilango, S. Nagaraj, M. Dong, et al. Predicting cellular responses to perturbation across diverse contexts with state. *BioRxiv*, pages 2025–06, 2025.

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

S. Bhattacharya, C. Gensbigler, S. Karim, and J. Lees. Discrete diffusion for single-cell gene expression modeling. *bioRxiv*, pages 2026–02, 2026.

H. Chang, H. Zhang, L. Jiang, C. Liu, and W. T. Freeman. MaskGIT: Masked generative image transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11315–11325, 2022.

H. Cui, C. Wang, H. Maan, K. Pang, F. Luo, N. Duan, and B. Wang. scgpt: toward building a foundation model for single-cell multi-omics using generative ai. *Nature methods*, 21(8):1470–1480, 2024.

A. Dixit, O. Parnas, B. Li, J. Chen, C. P. Fulco, L. Jerby-Arnon, N. D. Marjanovic, D. Dionne, T. Burks, R. Raychowdhury, et al. Perturb-seq: dissecting molecular circuits with scalable single-cell rna profiling of pooled genetic screens. *cell*, 167(7):1853–1866, 2016.

M. Dong, A. Adduri, D. Gautam, C. Carpenter, R. Shah, C. Ricci-Tam, Y. Kluger, D. P. Burke, and Y. H. Roohani. Stack: In-context learning of single-cell biology. *bioRxiv*, pages 2026–01, 2026.

G. Finak, A. McDavid, M. Yajima, J. Deng, V. Gersuk, A. K. Shalek, C. K. Slichter, H. W. Miller, M. J. McElrath, M. Prlic, et al. Mast: a flexible statistical framework for assessing transcriptional changes and characterizing heterogeneity in single-cell rna sequencing data. *Genome biology*, 16(1):278, 2015.

W. T. Fitch. Information and the single cell. *Current Opinion in Neurobiology*, 71:150–157, 2021.

S. Gandhi, F. Javadi, V. Svensson, U. Khan, M. G. Jones, J. Yu, D. Merico, H. Goodarzi, and N. Alidoust. Tahoe-x1: Scaling perturbation-trained single-cell foundation

models to 3 billion parameters. *bioRxiv*, pages 2025–10, 2025.

G. S. Gulati, S. S. Sikandar, D. J. Wesche, A. Manjunath, A. Bharadwaj, M. J. Berger, F. Ilagan, A. H. Kuo, R. W. Hsieh, S. Cai, et al. Single-cell transcriptional diversity is a hallmark of developmental potential. *Science*, 367 (6476):405–411, 2020.

M. Hao, J. Gong, X. Zeng, C. Liu, Y. Guo, X. Cheng, T. Wang, J. Ma, X. Zhang, and L. Song. Large-scale foundation model on single-cell transcriptomics. *Nature methods*, 21(8):1481–1491, 2024.

J. Hoffmann, S. Borgeaud, A. Mensch, E. Buchatskaya, T. Cai, E. Rutherford, D. de Las Casas, L. A. Hendricks, J. Welbl, A. Clark, T. Hennigan, E. Noland, K. Millican, G. van den Driessche, B. Damoc, A. Guy, S. Osindero, K. Simonyan, E. Elsen, J. W. Rae, O. Vinyals, and L. Sifre. Training compute-optimal large language models. *ArXiv*, abs/2203.15556, 2022. URL https://api.semanticscholar.org/CorpusID:247778764.

Z. Lin, H. Akin, R. Rao, B. Hie, Z. Zhu, W. Lu, N. Smetanin, R. Verkuil, O. Kabeli, Y. Shmueli, et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023.

Y. Lipman, R. T. Chen, H. Ben-Hamu, M. Nickel, and M. Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Y. Liu et al. CellTok: Early-fusion multimodal large language model for single-cell transcriptomics via tokenization. *bioRxiv*, pages 2025–10, 2025.

R. Lopez, J. Regier, M. Cole, M. Jordan, and N. Yosef. Deep generative modeling for single-cell transcriptomics. *Nature Methods*, 15, 12 2018a. doi: 10.1038/s41592-018-0229-2.

R. Lopez, J. Regier, M. B. Cole, M. I. Jordan, and N. Yosef. Deep generative modeling for single-cell transcriptomics. *Nature methods*, 15(12):1053–1058, 2018b.

M. Lotfollahi, F. A. Wolf, and F. J. Theis. scgen predicts single-cell perturbation responses. *Nature methods*, 16 (8):715–721, 2019.

M. Lotfollahi, M. Naghipourfar, F. J. Theis, and F. A. Wolf. Conditional out-of-distribution generation for unpaired data using transfer vae. *Bioinformatics*, 36 (Supplement_2):i610–i617, 2020.

M. Lotfollahi, A. Klimovskaia Susmelj, C. De Donno, L. Hetzel, Y. Ji, I. L. Ibarra, S. R. Srivatsan, M. Naghipourfar, R. M. Daza, B. Martin, et al. Predicting cellular responses to complex perturbations in high-throughput screens. *Molecular systems biology*, 19 (6):e11517, 2023.

A. Lou, C. Meng, and S. Ermon. Discrete diffusion modeling by estimating the ratios of the data distribution. In *International Conference on Machine Learning*, 2023. URL https://api.semanticscholar.org/CorpusID:264451832.

E. Luo, M. Hao, L. Wei, and X. Zhang. scdiffusion: conditional generation of high-quality single-cell data using diffusion model. *Bioinformatics*, 40, 2024. URL https://api.semanticscholar.org/CorpusID:266844972.

E. Z. Macosko, A. Basu, R. Satija, J. Nemesh, K. Shekhar, M. Goldman, I. Tirosh, A. R. Bialas, N. Kamitaki, E. M. Martersteck, et al. Highly parallel genome-wide expression profiling of individual cells using nanoliter droplets. *Cell*, 161(5):1202–1214, 2015.

F. Mentzer, D. C. Minnen, E. Agustsson, and M. Tschannen. Finite scalar quantization: Vq-vae made simple. *ArXiv*, abs/2309.15505, 2023. URL https://api.semanticscholar.org/CorpusID:263153393.

L. Oesinghaus, S. Becker, L. Vornholz, E. Papalexi, J. Pangallo, A. A. Moinfar, J. Liu, A. La Fleur, M. Shulman, S. Marrujo, et al. A single-cell cytokine dictionary of human peripheral blood. *bioRxiv*, 2025.

J. Ou, S. Nie, K. Xue, F. Zhu, J. Sun, Z. Li, and C. Li. Your absorbing discrete diffusion secretly models the conditional distributions of clean data. In *International Conference on Learning Representations (ICLR)*, 2025.

G. Palla, S. Babu, P. Dibaeinia, J. D. Pearce, D. Li, A. A. Khan, T. Karaletsos, and J. M. Tomczak. Scalable single-cell gene expression generation with latent diffusion models. *ArXiv*, abs/2511.02986, 2025. URL https://api.semanticscholar.org/CorpusID:282758604.

A. Palma, T. Richter, H. Zhang, M. Lubetzki, A. Tong, A. Dittadi, and F. J. Theis. Multi-modal and multi-attribute generation of single cells with cfgen. In *International Conference on Learning Representations*, 2024. URL https://api.semanticscholar.org/CorpusID:271218151.

W. Peebles and S. Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023.

A. Razavi, A. van den Oord, and O. Vinyals. Generating diverse high-fidelity images with VQ-VAE-2. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

J. M. Replogle, R. A. Saunders, A. N. Pogson, J. A. Hussmann, A. Lenail, A. Guna, L. Mascibroda, E. J. Wagner, K. Adelman, G. Lithwick-Yanai, et al. Mapping information-rich genotype-phenotype landscapes with genome-scale perturb-seq. *Cell*, 185(14):2559–2575, 2022.

R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022.

S. S. Sahoo, M. Arriola, Y. Schiff, A. Gokaslan, E. Marroquin, J. T. Chiu, A. Rush, and V. Kuleshov. Simple and effective masked diffusion language models. *ArXiv*, abs/2406.07524, 2024. URL `https://api.semanticscholar.org/CorpusID:270380319`.

J. Shi, K. Han, Z. Wang, A. Doucet, and M. Titsias. Simplified and generalized masked diffusion for discrete data. *Advances in neural information processing systems*, 37: 103131–103167, 2024.

C. V. Theodoris, L. Xiao, A. Chopra, M. D. Chaffin, Z. R. Al Sayed, M. C. Hill, H. Mantineo, E. M. Brydon, Z. Zeng, X. S. Liu, et al. Transfer learning enables predictions in network biology. *Nature*, 618(7965):616–624, 2023.

A. van den Oord, O. Vinyals, and K. Kavukcuoglu. Neural discrete representation learning. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

C. Wang, M. Karimzadeh, N. G. Ravindra, L. R. Bounds, N. Alerasool, A. C. Huang, S. Ma, D. R. Gulbranson, H. Cui, Y. Lee, et al. X-cell: Scaling causal perturbation prediction across diverse cellular contexts via diffusion language models. *bioRxiv*, pages 2026–03, 2026a.

M. Wang, C. Chen, G. Jiang, Z. Ren, C. Zhao, L. Shi, and Y. Ma. Scdiva: Masked discrete diffusion for joint modeling of single-cell identity and expression. *arXiv preprint arXiv:2602.03477*, 2026b.

D. R. Wong, M. Piper, J. Qiao, M. Russo, P. Jean, D.-A. Clevert, J. D. Arroyo, and E. Pashos. Phenotypic reversion and target prioritization for cellular inflammation via representation learning with foundation models. *bioRxiv*, pages 2026–03, 2026.

T. Zeng and H. Dai. Single-cell rna sequencing-based computational analysis to describe disease heterogeneity. *Frontiers in Genetics*, 10:629, 2019.

H. Zhang, G.-H. Yuan, C. Yuan, T. Xu, T. Bian, H. Cheng, W. Huang, D. Zhao, and Y. Rong. Lingshu-cell: A generative cellular world model for transcriptome modeling toward virtual cells. *arXiv preprint arXiv:2603.25240*, 2026.

A. Hyperparameters and training configuration

Tables 2 and 3 list the hyperparameters used for the main-paper VAE backbone and prior models. The same configuration is shared across Parse 1M and Replogle, with two dataset-specific differences: (i) the perturbation conditioning source — ESM2-3B protein-language-model embeddings keyed by the knocked-out gene on Replogle, and a learnable per-cytokine table on Parse 1M (cytokine ligands have no direct protein-sequence analog) — and (ii) the train/holdout split definitions, matched to scLDM in both cases. Both datasets use additive cell-type conditioning on the adaLN-Zero path.

Table 2. **Stage 1: `ExpressionVAE` hyperparameters.** Shared across both datasets and all four output heads. The Gaussian-latent variant referenced in Section 4.3 swaps the FSQ bottleneck for a Gaussian-posterior bottleneck with the parameters in the lower block; all other settings are unchanged.

Latent sequence length L	64
FSQ codebook size K	512
Encoder / decoder	Cross-attention transformer
Gene input embedding	MLP over $\log_1 p$ count + learned gene-identity tokens
Optimiser	AdamW
Learning rate	6×10^{-4}
Schedule	Cosine
Warm-up steps	2,500
Training iterations	75,000
Batch size	128
Gaussian-latent variant: per-token latent dim	8
Gaussian-latent variant: β_{KL}	2×10^{-6}

Table 3. **Stage 2: prior hyperparameters.** Shared across the autoregressive (AR) and masked-discrete-diffusion (MDLM) priors. The DiT backbone is identical (same depth, width, head count, parameter budget) so any gap in Section 4.1 reflects the prior family rather than capacity.

Backbone	DiT (Diffusion Transformer)
Conditioning	adaLN-Zero on perturbation + cell-type embedding
Model dimension d_{model}	384
Attention heads	8
Layers	6
Feed-forward dimension	1,536
Optimiser	AdamW
Learning rate	2×10^{-4}
Schedule	Cosine
Warm-up steps	3,000
Gradient clipping (max-norm)	1.0
Training iterations	75,000
Batch size	128
MDLM: masking schedule	absorbing, $\alpha_t = 1 - t$
MDLM: t_ϵ	5×10^{-3}
MDLM: inference denoising steps	32

B. Codebook size and latent-token-count ablation

We swept the FSQ latent sequence length $L \in \{64, 128\}$ against the codebook size $K \in \{512, 1024\}$ on the Replogle (HepG2) holdout, paired with the AR-FSQ-MSE and MDLM-FSQ-MSE configurations from the main paper. We additionally report AR-FSQ-NB, MDLM-FSQ-NB, and a Flow-Gaussian-MSE configuration as neighbouring anchor points in the design space. Results are in Table 4.

The numbers in Table 4 were obtained on a different gene-set than the main-paper results and are therefore not directly

comparable to Table 1; they are, however, internally consistent across rows. Based on this ablation we carried forward $L = 64$ and $K = 512$ to the main experiments.

Table 4. **Codebook size and latent-token-count ablation on the Replogle (HepG2) holdout.** Bold = best per column.

Run	PDS ↑	ovl@N ↑	PR-AUC ↑	Sp-sig ↑	W_2 ↓	MMD^2 ↓	FD ↓
AR-FSQ-MSE L=128, K=1024	0.7373	0.2800	0.2225	0.7519	**10.02**	**0.0706**	56.26
AR-FSQ-MSE L=128, K=512	0.7398	**0.2914**	**0.2301**	0.7499	10.17	0.0734	57.28
AR-FSQ-MSE L=64, K=1024	0.7474	0.2773	0.2196	0.7563	10.06	0.0720	56.70
AR-FSQ-MSE L=64, K=512	0.7510	0.2834	0.2177	**0.7856**	10.06	0.0716	57.19
AR-FSQ-NB	**0.7594**	0.2790	0.2224	0.7678	12.35	0.1514	116.30
MDLM-FSQ-MSE L=128, K=1024	0.7030	0.2750	0.2071	0.7265	10.18	0.0791	59.31
MDLM-FSQ-MSE L=128, K=512	0.7147	0.2790	0.2135	0.7440	10.25	0.0776	57.75
MDLM-FSQ-MSE L=64, K=1024	0.7126	0.2692	0.2064	0.7495	10.19	0.0771	57.80
MDLM-FSQ-MSE L=64, K=512	0.7169	0.2816	0.2161	0.7344	10.27	0.0777	60.17
MDLM-FSQ-NB	0.7259	0.2728	0.2112	0.7406	12.49	0.1580	118.57
Flow-Gaussian-MSE	0.7514	0.2820	0.2204	0.7290	10.17	0.0707	**55.10**

C. Analysis of Evaluation Metrics

We evaluate perturbation prediction across two complementary families of metrics: *differential expression (DE) metrics*, which assess how well a model reproduces the statistical signature of transcriptional change, and *distribution metrics*, which assess the fidelity of the full predicted cell population. We find that certain design choices lead to some statistical metrics generating NaN values for some perturbations. In Table 1 we ignore these NaN values or take them as 0.
To ensure fair comparison, we use the same training/test split as scLDM (Palla et al., 2025). Metrics like PR-AUC, Sp-sig, and P-Δ are not reported by (Palla et al., 2025) so we use the values reported as is from (Adduri et al., 2025). We also note that the latter train/test split is not matched in this case.

In the following, we describe each metric, its mathematical definition.

C.1. Differential Expression Metrics

All DE based metrics share a common upstream computation: for each perturbation p we derive a *pseudobulk* summary, compared against the control population of that perturbation.

Pseudobulk aggregation. Given a matrix of $\log_{1\text{p}}$-normalised expression values $X^{(p)} \in \mathbb{R}^{N_p \times G}$ (cells $\times$ genes) for perturbation p, and the analogous control matrix $X^{(\text{ctrl})} \in \mathbb{R}^{N_c \times G}$, we compute the geometric pseudobulk in count space as

$$\bar{x}_g^{(p)} = \operatorname{expm1}\Big(\tfrac{1}{N_p}\sum_{i=1}^{N_p} X_{ig}^{(p)}\Big), \qquad \bar{x}_g^{(\text{ctrl})} = \operatorname{expm1}\Big(\tfrac{1}{N_c}\sum_{i=1}^{N_c} X_{ig}^{(\text{ctrl})}\Big). \tag{1}$$

Taking the mean in log space and back-transforming via $\operatorname{expm1}(z) = e^z - 1$ is equivalent to computing the geometric mean of the original (pre-log) counts.

Log_2 fold-change.

$$\text{LFC}_g^{(p)} = \log_2 \frac{\bar{x}_g^{(p)} + \varepsilon}{\bar{x}_g^{(\text{ctrl})} + \varepsilon}, \tag{2}$$

where $\varepsilon = 0$ by default; genes with a zero control mean therefore yield $\pm\infty$ and are treated as such throughout the pipeline.

Per-perturbation FDR. Differential expression is tested per gene with a two-sided Mann–Whitney U test applied to the raw ($\log_{1\text{p}}$) cell vectors of perturbation p and controls. p-values are corrected to false discovery rates (FDR) using the Benjamini–Hochberg (BH) procedure, applied *independently within each perturbation* (i.e. across the G gene p-values for that perturbation).

C.1.1. PRECISION–RECALL AUC (PR-AUC)

PR-AUC quantifies how well the *predicted* FDR scores rank the truly DE genes. For each perturbation p:

$$\ell_g = \mathbf{1}\Big[\mathrm{FDR}_g^{(p),\mathrm{real}} < 0.05\Big], \tag{3}$$

$$s_g = -\log_{10}\Big(\mathrm{clip}\big(\mathrm{FDR}_g^{(p),\mathrm{pred}},\ 10^{-10},\ 1\big)\Big), \tag{4}$$

and the per-perturbation area under the precision–recall curve (AP) is computed with `sklearn.metrics.average_precision_score`$(\boldsymbol{\ell}, \mathbf{s})$. Missing predicted FDR values are filled with 1.0 (least confident), and we clip to $[10^{-10}, 1]$ before taking the log to avoid numerical issues. We report the mean AP across perturbations.

C.1.2. SPEARMAN OF SIGNIFICANT-GENE COUNTS (SP-SIG)

Sp-sig measures across-perturbation agreement in how many genes are called DE, rather than their identities or magnitudes. Let $n_p^{(\mathrm{real})} = |\{g : \mathrm{FDR}_g^{(p),\mathrm{real}} < 0.05\}|$ and $n_p^{(\mathrm{pred})} = |\{g : \mathrm{FDR}_g^{(p),\mathrm{pred}} < 0.05\}|$. Then

$$\text{Sp-sig} = \rho_S\Big(\big(n_p^{(\mathrm{real})}\big)_p,\ \big(n_p^{(\mathrm{pred})}\big)_p\Big), \tag{5}$$

where ρ_S denotes the Spearman rank correlation computed over all perturbations. A model that produces zero significant genes everywhere yields a constant predicted vector; we follow the convention of returning `NaN` in this degenerate case and use `nanmean` when averaging over seeds.

C.1.3. SPEARMAN OF LOG$_2$FC OVER SIGNIFICANT GENES (SP-LFC)

Sp-LFC measures per-perturbation agreement in the *direction and magnitude* of DE, restricted to genes that are truly significant. For each perturbation p, let $\mathcal{S}_p = \{g : \mathrm{FDR}_g^{(p),\mathrm{real}} < 0.05\}$ be the real-significant gene set.

$$\text{Sp-LFC}_p = \rho_S\Big(\big(\mathrm{LFC}_g^{(p),\mathrm{real}}\big)_{g\in\mathcal{S}_p},\ \big(\mathrm{LFC}_g^{(p),\mathrm{pred}}\big)_{g\in\mathcal{S}_p}\Big), \tag{6}$$

where predicted LFC values for genes absent from the predicted DE table are filled with 0.0. We report the mean over perturbations. Note that both the gene set $\mathcal{S}_p$ and the LFC values are defined solely from *real* data; the predicted LFC is only evaluated at those positions.

C.1.4. PERTURBATION DISCRIMINATION SCORE (PDS)

The PDS measures the *specificity* of predicted perturbation effects: does the predicted effect for perturbation p match the real effect of p more closely than the real effects of all other perturbations?

Let $\boldsymbol{\delta}^{(p),\mathrm{real}} = \bar{\mathbf{x}}^{(p)} - \bar{\mathbf{x}}^{(\mathrm{ctrl})}$ and $\boldsymbol{\delta}^{(p),\mathrm{pred}} = \hat{\mathbf{x}}^{(p)} - \bar{\mathbf{x}}^{(\mathrm{ctrl})}$ denote the real and predicted perturbation effects (pseudobulk deltas from control). For each perturbation p, we compute the L_1 distance between the predicted effect and every real effect:

$$d_{p,q} = \big\|\boldsymbol{\delta}^{(p),\mathrm{pred}} - \boldsymbol{\delta}^{(q),\mathrm{real}}\big\|_1, \quad q = 1, \ldots, P, \tag{7}$$

then rank the P distances in ascending order and find the rank r_p of the true perturbation $q = p$.

$$\mathrm{PDS}_p = 1 - \frac{r_p}{P}, \tag{8}$$

so $\mathrm{PDS}_p = 1$ indicates the correct perturbation is the nearest neighbour, and $\mathrm{PDS}_p = 0.5$ is chance performance.

CRISPR vs. non-CRISPR data. For CRISPR knockdown experiments, the target gene itself exhibits a strong direct knockdown signal ($\sim 95\%$ reduction) that would trivially dominate the L_1 distance and inflate PDS for any model that simply predicts a reduction at the target locus. We therefore exclude the target gene from the distance calculation when evaluating on CRISPR perturbation data, following Adduri et al. (2025):

$$d_{p,q} = \big\|\boldsymbol{\delta}_{\setminus\{g_p\}}^{(p),\mathrm{pred}} - \boldsymbol{\delta}_{\setminus\{g_p\}}^{(q),\mathrm{real}}\big\|_1, \tag{9}$$

where g_p denotes the target gene of perturbation p and the subscript $\setminus\{g_p\}$ denotes the remaining gene dimensions. For non-CRISPR perturbations (cytokines, small molecules) there is no analogous single-gene confound, so no gene is excluded.

C.2. Distribution-Level Metrics

Beyond DE summary statistics, we evaluate fidelity of the full predicted cell distribution against ground-truth cells for each perturbation. All distribution metrics operate in the top-30 PCA components of the gene expression space; the PCA basis is fitted on the ground-truth cells and used to project both real and predicted populations. We match this to the design details mentioned in (Palla et al., 2025).

C.2.1. Maximum Mean Discrepancy (MMD^2)

The squared MMD between predicted cells $\hat{\mathcal{P}}$ and real cells $\mathcal{Q}$ under an RBF kernel $k(\mathbf{x}, \mathbf{y}) = \exp(-\|\mathbf{x} - \mathbf{y}\|^2/\sigma^2)$ is

$$\mathrm{MMD}^2(\hat{\mathcal{P}}, \mathcal{Q}) = \mathbb{E}[k(\mathbf{x}, \mathbf{x}')] + \mathbb{E}[k(\mathbf{y}, \mathbf{y}')] - 2\,\mathbb{E}[k(\mathbf{x}, \mathbf{y})], \tag{10}$$

where $\mathbf{x}, \mathbf{x}' \sim \hat{\mathcal{P}}$ and $\mathbf{y}, \mathbf{y}' \sim \mathcal{Q}$. We use the median heuristic to set σ on the real cells.

C.2.2. 2-Wasserstein Distance (W_2)

The 2-Wasserstein distance is defined as the minimum-cost coupling between the two distributions under the squared-L_2 ground metric:

$$W_2(\hat{\mathcal{P}}, \mathcal{Q}) = \Big(\min_{\gamma \in \Gamma(\hat{\mathcal{P}}, \mathcal{Q})} \mathbb{E}_{(\mathbf{x},\mathbf{y}) \sim \gamma} \big[\|\mathbf{x} - \mathbf{y}\|^2 \big] \Big)^{1/2}, \tag{11}$$

where $\Gamma(\hat{\mathcal{P}}, \mathcal{Q})$ is the set of all couplings. We solve the discrete OT problem exactly via the Earth Mover's Distance (EMD) solver on subsampled populations (up to 5 000 cells), and report W_2 in PCA-30 space. Sinkhorn is an approximate version of the EMD solver.

C.2.3. Fréchet Distance (FD)

The Fréchet distance treats each distribution as a multivariate Gaussian and measures the distance between the fitted Gaussians:

$$\mathrm{FD} = \|\boldsymbol{\mu}_{\hat{P}} - \boldsymbol{\mu}_Q\|^2 + \mathrm{Tr}\Big(\boldsymbol{\Sigma}_{\hat{P}} + \boldsymbol{\Sigma}_Q - 2\big(\boldsymbol{\Sigma}_{\hat{P}} \boldsymbol{\Sigma}_Q\big)^{1/2} \Big), \tag{12}$$

where $(\cdot)^{1/2}$ denotes the matrix square root. We add $\varepsilon \mathbf{I}$ ($\varepsilon = 10^{-6}$) to each covariance matrix before computing the matrix square root to handle rank-deficient PCA components.

C.3. Implementation Notes and Cross-Library Validation

Table 5 summarises the metrics used in our main evaluation, grouped by the aspect of perturbation response they assess.

Table 5. Summary of evaluation metrics. DE metrics share the same upstream pseudobulk and per-perturbation BH-FDR computation; distribution metrics operate in PCA-30 space fitted on ground-truth cells. The last column gives each metric's inference-time sampling axis (Section 4.2): *variance-sensitive* metrics reward reproducing the within-perturbation distribution and full per-gene profile (favoring sampling heads), *mean-sensitive* metrics read a clean mean over strongly-DE genes (favoring deterministic heads). PDS is variance-sensitive despite being a pseudobulk metric: its L_1 is dominated by the low-count bulk, unlike *Pearson* Δ, a correlation dominated by the strong-DE genes.

Metric	Aspect assessed	Sampling axis	Opt.
MMD^2	Distribution fidelity	Variance-sensitive	↓
W_2	Distribution fidelity	Variance-sensitive	↓
FD	Distribution fidelity	Variance-sensitive	↓
PR-AUC	DE detection (FDR)	Variance-sensitive	↑
Sp-sig	DE detection (sig. count)	Variance-sensitive	↑
PDS	Specificity (full-vector L_1)	Variance-sensitive	↑
Sp-LFC	Effect magnitude (DE-gene rank)	Mean-sensitive	↑
overlap@N	Top-DE-gene recovery	Mean-sensitive	↑
Pearson Δ	Effect correlation (full vector)	Mean-sensitive	↑

Table 6. **Full prior × head ablation on Distribution metrics for Parse 1M and Replogle (HepG2).** We add results from running 4-seed pooled mean ± SE runs. Bold = best in column within each dataset.

Dataset	Prior	Head	$W_2 \downarrow$	$MMD^2 \downarrow$	FD ↓
Parse 1M	AR	ce-quantile	$11.978_{\pm 0.011}$	$\mathbf{0.0039}_{\pm 0.0001}$	$\mathbf{1.78}_{\pm 0.02}$
		nb-sample	$11.952_{\pm 0.009}$	$0.0081_{\pm 0.0002}$	$3.31_{\pm 0.05}$
		hurdle	$12.045_{\pm 0.007}$	$0.0114_{\pm 0.0003}$	$5.00_{\pm 0.10}$
		mse	$22.080_{\pm 0.108}$	$0.6462_{\pm 0.0047}$	$375.41_{\pm 4.54}$
	flow	ce-quantile	$12.243_{\pm 0.008}$	$0.0246_{\pm 0.0002}$	$11.06_{\pm 0.15}$
		nb-sample	$12.600_{\pm 0.007}$	$0.0460_{\pm 0.0005}$	$20.48_{\pm 0.24}$
		hurdle	$11.920_{\pm 0.005}$	$0.0073_{\pm 0.0001}$	$2.94_{\pm 0.03}$
		mse	$\mathbf{11.893}_{\pm 0.005}$	$0.0111_{\pm 0.0003}$	$4.17_{\pm 0.10}$
	MDLM	ce-quantile	$11.997_{\pm 0.004}$	$0.0047_{\pm 0.0001}$	$2.13_{\pm 0.04}$
		nb-sample	$11.969_{\pm 0.007}$	$0.0081_{\pm 0.0001}$	$3.50_{\pm 0.06}$
		hurdle	$12.083_{\pm 0.010}$	$0.0129_{\pm 0.0002}$	$5.46_{\pm 0.08}$
		mse	$22.152_{\pm 0.073}$	$0.6542_{\pm 0.0031}$	$378.10_{\pm 2.81}$
Replogle	AR	ce-quantile	$\mathbf{7.864}_{\pm 0.010}$	$\mathbf{0.0089}_{\pm 0.0003}$	$4.54_{\pm 0.08}$
		nb-sample	$8.001_{\pm 0.010}$	$0.0095_{\pm 0.0003}$	$4.70_{\pm 0.09}$
		hurdle	$7.900_{\pm 0.010}$	$0.0130_{\pm 0.0004}$	$5.47_{\pm 0.11}$
		mse	$9.032_{\pm 0.017}$	$0.0987_{\pm 0.0007}$	$35.06_{\pm 0.24}$
	flow	ce-quantile	$8.049_{\pm 0.008}$	$0.0090_{\pm 0.0001}$	$\mathbf{4.34}_{\pm 0.08}$
		nb-sample	$8.217_{\pm 0.007}$	$0.0192_{\pm 0.0003}$	$6.20_{\pm 0.09}$
		hurdle	$7.885_{\pm 0.005}$	$0.0114_{\pm 0.0002}$	$4.93_{\pm 0.06}$
		mse	$8.971_{\pm 0.019}$	$0.0783_{\pm 0.0012}$	$27.14_{\pm 0.34}$
	MDLM	ce-quantile	$8.051_{\pm 0.020}$	$0.0133_{\pm 0.0004}$	$5.36_{\pm 0.15}$
		nb-sample	$8.121_{\pm 0.014}$	$0.0126_{\pm 0.0004}$	$5.29_{\pm 0.10}$
		hurdle	$8.042_{\pm 0.008}$	$0.0172_{\pm 0.0005}$	$6.33_{\pm 0.07}$
		mse	$9.091_{\pm 0.021}$	$0.1052_{\pm 0.0011}$	$35.70_{\pm 0.38}$

D. Continous latent spaces

Upon the results presented in section C, we were curious if this followed continous latent spaces as well. The community has not released these versions among existing benchmarks, like ce-quantile, and hurdle with flow. Other ablations like mse and nb have been published across multiple studies. Here are the results for the full ablations on all the discrete and continous latent spaces. We also want to point out that while the flow model does better on the Parse 1M dataset - the same trend does not follow for the replogle dataset, and therefore it maybe just an artifact of the dataset. We hope to add further analysis in this study by expanding it to more datasets.

Table 7. **All prior × head ablations across cell-eval metrics**

Dataset	Prior	Head	disc_l1 ↑	ovl@N ↑	Sp-LFC ↑	PR-AUC ↑	MAE-l2 ↓	P-△ ↑	MAE-pb ↓	Sp-sig ↑
Parse 1M	AR	ce-quantile	$0.962_{\pm 0.003}$	$0.409_{\pm 0.005}$	$0.613_{\pm 0.015}$	$0.505_{\pm 0.002}$	$0.609_{\pm 0.001}$	$0.723_{\pm 0.000}$	$\mathbf{0.023}_{\pm 0.000}$	$0.940_{\pm 0.007}$
		nb-sample	$0.948_{\pm 0.003}$	$0.407_{\pm 0.005}$	$0.655_{\pm 0.016}$	$0.498_{\pm 0.004}$	$0.612_{\pm 0.001}$	$0.762_{\pm 0.000}$	$0.027_{\pm 0.000}$	$0.949_{\pm 0.005}$
		hurdle	$0.916_{\pm 0.002}$	$0.312_{\pm 0.004}$	$0.524_{\pm 0.012}$	$0.432_{\pm 0.005}$	$0.732_{\pm 0.002}$	$0.723_{\pm 0.000}$	$0.030_{\pm 0.000}$	$0.939_{\pm 0.003}$
		mse	$0.632_{\pm 0.004}$	$0.140_{\pm 0.005}$	$0.618_{\pm 0.003}$	$0.157_{\pm 0.002}$	$0.700_{\pm 0.001}$	$0.747_{\pm 0.000}$	$0.324_{\pm 0.000}$	$0.926_{\pm 0.007}$
	flow	ce-quantile	$0.811_{\pm 0.006}$	$0.236_{\pm 0.005}$	$0.496_{\pm 0.005}$	$0.353_{\pm 0.005}$	$0.804_{\pm 0.002}$	$0.630_{\pm 0.000}$	$0.040_{\pm 0.000}$	$0.924_{\pm 0.004}$
		nb-sample	$0.818_{\pm 0.003}$	$0.355_{\pm 0.005}$	$0.661_{\pm 0.005}$	$0.484_{\pm 0.005}$	$0.568_{\pm 0.001}$	$0.780_{\pm 0.000}$	$0.044_{\pm 0.000}$	$0.959_{\pm 0.004}$
		hurdle	$\mathbf{1.000}_{\pm 0.001}$	$\mathbf{0.448}_{\pm 0.004}$	$\mathbf{0.693}_{\pm 0.003}$	$\mathbf{0.557}_{\pm 0.004}$	$\mathbf{0.542}_{\pm 0.001}$	$0.832_{\pm 0.000}$	$0.025_{\pm 0.000}$	$0.961_{\pm 0.004}$
		mse	$0.988_{\pm 0.001}$	$0.371_{\pm 0.004}$	$0.684_{\pm 0.002}$	$0.127_{\pm 0.002}$	$0.549_{\pm 0.001}$	$\mathbf{0.841}_{\pm 0.000}$	$0.054_{\pm 0.000}$	$0.882_{\pm 0.005}$
	MDLM	ce-quantile	$0.964_{\pm 0.002}$	$0.405_{\pm 0.005}$	$0.577_{\pm 0.007}$	$0.497_{\pm 0.002}$	$0.602_{\pm 0.001}$	$0.716_{\pm 0.000}$	$\mathbf{0.023}_{\pm 0.000}$	$\mathbf{0.969}_{\pm 0.007}$
		nb-sample	$0.956_{\pm 0.003}$	$0.401_{\pm 0.005}$	$0.639_{\pm 0.017}$	$0.485_{\pm 0.004}$	$0.612_{\pm 0.002}$	$0.763_{\pm 0.000}$	$0.027_{\pm 0.000}$	$0.960_{\pm 0.006}$
		hurdle	$0.896_{\pm 0.004}$	$0.296_{\pm 0.004}$	$0.531_{\pm 0.021}$	$0.429_{\pm 0.005}$	$0.746_{\pm 0.003}$	$0.724_{\pm 0.000}$	$0.031_{\pm 0.000}$	$0.924_{\pm 0.005}$
		mse	$0.624_{\pm 0.003}$	$0.144_{\pm 0.005}$	$0.646_{\pm 0.008}$	$0.141_{\pm 0.003}$	$0.715_{\pm 0.002}$	$0.733_{\pm 0.000}$	$0.325_{\pm 0.000}$	$0.901_{\pm 0.006}$
Replogle	AR	ce-quantile	$0.740_{\pm 0.004}$	$0.164_{\pm 0.003}$	$0.513_{\pm 0.012}$	$0.282_{\pm 0.001}$	$0.840_{\pm 0.003}$	$0.310_{\pm 0.000}$	$0.109_{\pm 0.000}$	$0.793_{\pm 0.005}$
		nb-sample	$\mathbf{0.763}_{\pm 0.003}$	$0.170_{\pm 0.002}$	$0.480_{\pm 0.009}$	$0.295_{\pm 0.001}$	$0.829_{\pm 0.002}$	$0.307_{\pm 0.000}$	$\mathbf{0.106}_{\pm 0.000}$	$0.789_{\pm 0.004}$
		hurdle	$0.745_{\pm 0.002}$	$0.213_{\pm 0.001}$	$0.461_{\pm 0.006}$	$\mathbf{0.298}_{\pm 0.001}$	$0.850_{\pm 0.002}$	$0.308_{\pm 0.000}$	$0.108_{\pm 0.000}$	$0.784_{\pm 0.003}$
		mse	$0.716_{\pm 0.003}$	$\mathbf{0.255}_{\pm 0.002}$	$0.508_{\pm 0.009}$	$0.230_{\pm 0.002}$	$0.839_{\pm 0.002}$	$\mathbf{0.381}_{\pm 0.000}$	$0.129_{\pm 0.000}$	$0.793_{\pm 0.004}$
	flow	ce-quantile	$0.739_{\pm 0.001}$	$0.169_{\pm 0.005}$	$0.463_{\pm 0.007}$	$0.283_{\pm 0.002}$	$0.851_{\pm 0.001}$	$0.295_{\pm 0.000}$	$0.109_{\pm 0.000}$	$0.787_{\pm 0.007}$
		nb-sample	$0.761_{\pm 0.001}$	$0.165_{\pm 0.003}$	$0.490_{\pm 0.006}$	$0.286_{\pm 0.001}$	$0.830_{\pm 0.001}$	$0.301_{\pm 0.000}$	$0.114_{\pm 0.000}$	$0.772_{\pm 0.006}$
		hurdle	$0.758_{\pm 0.001}$	$0.210_{\pm 0.001}$	$0.501_{\pm 0.005}$	$0.297_{\pm 0.001}$	$0.829_{\pm 0.002}$	$0.309_{\pm 0.000}$	$0.107_{\pm 0.000}$	$0.787_{\pm 0.005}$
		mse	$0.731_{\pm 0.003}$	$0.251_{\pm 0.001}$	$0.534_{\pm 0.007}$	$0.218_{\pm 0.001}$	$\mathbf{0.824}_{\pm 0.004}$	$0.367_{\pm 0.000}$	$0.125_{\pm 0.001}$	$0.778_{\pm 0.001}$
	MDLM	ce-quantile	$0.715_{\pm 0.003}$	$0.158_{\pm 0.002}$	$0.439_{\pm 0.004}$	$0.275_{\pm 0.002}$	$0.857_{\pm 0.003}$	$0.297_{\pm 0.000}$	$0.110_{\pm 0.000}$	$0.774_{\pm 0.005}$
		nb-sample	$0.756_{\pm 0.002}$	$0.192_{\pm 0.002}$	$0.495_{\pm 0.007}$	$0.297_{\pm 0.002}$	$0.825_{\pm 0.002}$	$0.318_{\pm 0.000}$	$0.108_{\pm 0.000}$	$\mathbf{0.799}_{\pm 0.005}$
		hurdle	$0.728_{\pm 0.001}$	$0.216_{\pm 0.001}$	$0.490_{\pm 0.009}$	$0.290_{\pm 0.001}$	$0.831_{\pm 0.002}$	$0.308_{\pm 0.000}$	$0.110_{\pm 0.000}$	$0.783_{\pm 0.004}$
		mse	$0.698_{\pm 0.004}$	$0.248_{\pm 0.001}$	$\mathbf{0.540}_{\pm 0.007}$	$0.219_{\pm 0.001}$	$0.829_{\pm 0.002}$	$0.362_{\pm 0.000}$	$0.130_{\pm 0.000}$	$0.793_{\pm 0.007}$