Scaling Laws for Masked-Reconstruction Transformers on Single-Cell Transcriptomics

Anonymous Authors

February 2026

Abstract

Neural scaling laws—power-law relationships between loss, model size, and data—have been extensively documented for language and vision transformers, yet their existence in single-cell genomics remains largely unexplored. We present the first systematic study of scaling behaviour for masked-reconstruction transformers trained on single-cell RNA sequencing (scRNA-seq) data. Using expression profiles from the CELLxGENE Census, we study a canonical regime with 512 highly variable genes and 200,000 cells, alongside an exploratory comparison regime with 1,024 genes and 10,000 cells. Across seven model sizes spanning three orders of magnitude in parameter count (533 to 3.4×10^8 parameters), we fit the parametric scaling law $L = aP^{-\alpha} + c$ to validation mean squared error (MSE). The canonical $V = 512$, $D = 200$k regime exhibits clear power-law scaling on validation loss, and held-out test evaluation follows the same qualitative trend. By contrast, the original $V = 1{,}024$, $D = 10$k comparison does not provide a clean causal test of data scarcity because vocabulary size, dataset size, and training budgets all differ simultaneously; we therefore treat it as exploratory rather than definitive. We additionally report matched-V follow-up analyses, including a fixed-$V = 512$ data-size sweep, held-out test-set scaling, and an empirical check of cross-gene residual heterogeneity. Under a homoscedastic Gaussian approximation, the asymptotic floor in the canonical regime corresponds to approximately 2.3 bits of *irreducible uncertainty remaining* per masked gene position, not to a universal biological constant. We discuss implications for the design of single-cell foundation models and outline the additional matched sweeps and likelihood-based objectives needed to turn this preliminary quantity into a rigorous transcriptomic entropy estimate.

1 Introduction

A central empirical finding in deep learning is that, under appropriate conditions, the loss of a neural network follows a power law as model size, dataset size, or training compute increases [1, 2]. These *scaling laws* have proven remarkably consistent across modalities: they hold for autoregressive language models [1, 3], vision transformers [4], and multimodal systems [5]. Beyond their theoretical interest, scaling laws have immediate practical value: they enable researchers to predict the performance of expensive large-scale training runs from cheap small-scale experiments, thereby guiding resource allocation and architectural choices.

Single-cell RNA sequencing (scRNA-seq) has become a cornerstone of modern biology, generating expression profiles for millions of individual cells across diverse tissues, species, and disease states [8]. In response, the community has developed increasingly large *foundation models* for single-cell data, including scVI [10], scGPT [12], Geneformer [13], scBERT [14], and scFoundation [15]. These models are typically evaluated on downstream tasks such as cell-type annotation, batch correction, or perturbation prediction, but—in contrast to the language modelling literature—the fundamental

question of how pretraining loss scales with model size and data volume has received almost no attention.

This gap matters for several reasons. First, without scaling laws, practitioners cannot determine whether a larger model will yield meaningful improvements or whether the current bottleneck is data, compute, or intrinsic noise. Second, the asymptotic loss floor c in a scaling law of the form $L = aP^{-\alpha} + c$ provides an empirical upper bound on the *reducible* error, which can be related to the intrinsic entropy of the data-generating process [1, 3]. For transcriptomics, such an estimate would quantify the fundamental information content of gene expression measurements—a quantity of independent biological interest.

In this work, we take a first step towards establishing scaling laws for single-cell transcriptomics. We train masked-reconstruction transformers—a pretraining paradigm analogous to masked language modelling [16]—on scRNA-seq data drawn from the CELLxGENE Census [9], varying the model size across seven presets that span from 533 to 3.4×10^8 trainable parameters. We study two regimes that differ in both gene vocabulary size and dataset size, fit power-law models to the resulting loss curves, and analyse the conditions under which scaling behaviour emerges or breaks down.

Contributions.

1. We present, to our knowledge, the first systematic investigation of neural scaling laws for masked-reconstruction pretraining on scRNA-seq data.
2. We demonstrate that power-law scaling emerges in a canonical $V = 512$, $D = 200$k regime, with held-out test evaluation following the same qualitative ordering across model sizes.
3. We show that the originally submitted $V = 1{,}024$, $D = 10$k comparison should be interpreted as exploratory because it is confounded in both dataset size and vocabulary size and mixes heterogeneous training budgets.
4. We add controlled follow-up analyses from existing runs, including a fixed-$V = 512$ data-size sweep, held-out test-set evaluation, and a representative cross-gene residual heterogeneity diagnostic.
5. We identify the key experimental ingredients—matched V/D sweeps, likelihood-based objectives, and comprehensive compute accounting—needed to progress from loss-vs-parameters curves to transcriptomic entropy estimates.

2 Related Work

Neural scaling laws. Hestness et al. [2] first documented power-law scaling of test error with dataset size across several domains. Kaplan et al. [1] established that language model cross-entropy loss follows $L(P) = (P_c/P)^{\alpha_P}$ over several orders of magnitude in parameter count P, with $\alpha_P \approx 0.076$ for autoregressive transformers. Importantly, they found that scaling with model size, data size, and compute each follows its own power law, and that training can be either model-limited or data-limited depending on the regime. Hoffmann et al. [3] refined these findings, demonstrating that Kaplan *et al.*'s protocol over-allocates parameters relative to data and proposing compute-optimal (*Chinchilla*) scaling rules. Subsequent work has extended scaling laws to vision transformers [4], contrastive multimodal models [5], and various other architectures and tasks [6, 7].

Single-cell foundation models. The single-cell community has developed several large pretrained models. scVI [10] introduced variational autoencoders for scRNA-seq, and subsequent work extended this framework to multi-omic data [11]. More recently, transformer-based architectures have gained

prominence: scGPT [12] adopts a generative pretraining approach with gene tokens; Geneformer [13] pretrains on rank-ordered gene expression using a BERT-like masked objective; scBERT [14] similarly uses masked gene expression prediction; and scFoundation [15] scales to 100 million parameters trained on 50 million cells. While these works report downstream benchmark results, none systematically characterizes how pretraining loss scales with model size—the question we address here.

Masked autoencoders for genomics. Masked reconstruction is a well-established self-supervised pretraining strategy. In natural language processing, BERT [16] popularized masked token prediction. In computer vision, masked autoencoders (MAE) [17] demonstrated that masking high fractions of image patches leads to effective visual representations. For single-cell data, the masked reconstruction objective is a natural fit: by randomly masking a subset of gene expression values and training the model to reconstruct them from the remaining genes, the model learns co-expression structure and gene regulatory relationships. Our architecture follows this paradigm, using a permutation-invariant transformer encoder with a per-gene reconstruction head.

3 Data

3.1 Source and Preprocessing

All data are drawn from the CELLxGENE Census [9], a curated aggregation of publicly available single-cell RNA-seq datasets encompassing diverse human tissues and cell types. We construct two experimental regimes, each defined by a gene vocabulary size V (the number of highly variable genes retained) and a dataset size D (the number of cells):

- **Regime A** ($V = 512$, $D = 200{,}000$): 200,000 cells with 512 highly variable genes (HVGs), stored as `census_human.D200000.V512.log1p.h5ad`.
- **Regime B** ($V = 1{,}024$, $D = 10{,}000$): 10,000 cells with 1,024 HVGs, stored as `census_subset.D10000.V1024.log1p.h5ad`.

The preprocessing pipeline proceeds as follows. First, we load the expression matrix from the Census and select the expression layer with a valid library size (preferring raw counts where available). Zero-library cells are removed. Second, we select highly variable genes using the Seurat v3 method [18], which ranks genes by their variance-to-mean ratio after variance-stabilizing transformation. Third, we apply library-size normalization to a target sum of 10^4 followed by the $\log(1+x)$ transform. The resulting matrix is stored in float32. Finally, we split cells into training (90%), validation (5%), and test (5%) partitions using stratified sampling based on available cell-type annotations where possible, with random splitting as fallback.

3.2 Regime Comparison

The two regimes differ along two axes simultaneously: gene vocabulary size and dataset size. This confounding is a central limitation of the current study (see Section 9). Accordingly, the cross-regime comparison should be interpreted as an exploratory contrast rather than as a clean causal test of whether dataset size alone explains the observed behaviour. We therefore use Regime A as the paper's canonical scaling experiment and treat Regime B as a motivating negative-control-style observation whose interpretation requires matched follow-up sweeps. Regime A provides a data-rich environment ($D/V \approx 390$ cells per gene) while Regime B is comparatively data-starved ($D/V \approx 10$ cells per gene).

4 Model Architecture

4.1 Overview

We employ a permutation-invariant transformer encoder designed for set-structured inputs, where each element of the set corresponds to a gene. The architecture follows the standard transformer encoder [19] but omits positional encodings entirely, since the ordering of genes carries no inherent biological meaning. The model receives three inputs: gene identifiers (integer indices into a learned embedding table), observed expression values, and a binary mask indicating which values have been hidden.

4.2 Input Embedding

Each gene token is formed by summing two components:

1. A **gene identity embedding** $\mathbf{e}_g \in \mathbb{R}^d$, obtained by looking up the gene index in a learned embedding table $\mathbf{E} \in \mathbb{R}^{V \times d}$.
2. A **value projection** $\mathbf{v}_g = \mathbf{W}_v x_g + \mathbf{b}_v$, where $\mathbf{W}_v \in \mathbb{R}^{d \times 1}$ and x_g is the (possibly masked) expression value for gene g.

At masked positions, the value projection is replaced by a learned **mask token** $\mathbf{m} \in \mathbb{R}^d$, initialized to zero and updated during training. The resulting token representation for gene g is:

$$\mathbf{h}_g^{(0)} = \mathbf{e}_g + \begin{cases} \mathbf{v}_g & \text{if gene } g \text{ is observed} \\ \mathbf{m} & \text{if gene } g \text{ is masked.} \end{cases} \tag{1}$$

4.3 Transformer Encoder

The token representations $\{\mathbf{h}_g^{(0)}\}_{g=1}^{V}$ are processed by a stack of L standard transformer encoder layers [19], each comprising multi-head self-attention and a position-wise feed-forward network with GELU activation [20]. We use pre-layer normalization (Pre-LN) [21], which places layer normalization before (rather than after) the attention and feed-forward sub-layers, yielding more stable training dynamics at scale. The feed-forward hidden dimension is $4d$ by default.

4.4 Prediction Head and Pooling

The output of the encoder at each gene position is passed through a linear prediction head that maps $\mathbb{R}^d \to \mathbb{R}^1$, producing a scalar expression estimate $\hat{x}_g$ for each gene. The loss is computed only at masked positions (Section 5). For downstream applications requiring a cell-level representation, the model produces a pooled embedding $\mathbf{z} \in \mathbb{R}^d$ via mean-pooling over all gene tokens.

4.5 Model Sizes

To study scaling behaviour, we define seven size presets that span approximately five orders of magnitude in parameter count. Table 1 summarizes their configurations. Parameter counts are reported for the two vocabulary sizes used in our experiments; the difference arises from the gene embedding matrix $\mathbf{E} \in \mathbb{R}^{V \times d}$.

Table 1: Model size presets used in scaling experiments. Parameter counts are computed for gene vocabularies $V = 512$ and $V = 1{,}024$. d: model dimension; L: number of encoder layers; H: number of attention heads; FFN mult: feed-forward expansion factor.

Size	d	L	H	**FFN mult**	**Params** ($V = 512$)	**Params** ($V = 1{,}024$)
XXS	1	1	1	1	534	1,046
TINY	16	1	1	1	9,937	18,129
XS	64	2	4	4	132,993	165,761
S	128	4	8	4	859,137	924,673
M	512	6	8	4	19,178,497	19,440,641
L	1020	8	12	4	101,033,041	101,555,281
XL	1536	12	16	4	341,295,105	341,557,249

5 Training

5.1 Masked Reconstruction Objective

Following the masked language modelling paradigm [16], we randomly mask 15% of gene expression values in each training sample. The training objective is to minimize the mean squared error (MSE) between the model's predictions and the true expression values at masked positions only:

$$\mathcal{L} = \frac{1}{|\mathcal{M}|} \sum_{g \in \mathcal{M}} \left(\hat{x}_g - x_g\right)^2, \tag{2}$$

where $\mathcal{M}$ denotes the set of masked gene indices and $|\mathcal{M}|$ its cardinality. We also monitor masked mean absolute error (MAE) as a secondary metric.

5.2 Optimization

We use AdamW [22] with weight decay $\lambda = 0.01$ and gradient clipping at norm 1.0. For Regime A runs on GPU, the base learning rate is 3.125×10^{-5}, computed as $\mathrm{lr}_{\mathrm{base}} \cdot \mathrm{eff_batch}/256$ where $\mathrm{lr}_{\mathrm{base}} = 2.5 \times 10^{-4}$ and the effective batch size is 32 (physical batch of 4 with 8 gradient accumulation steps). All GPU runs use mixed-precision training (float16 autocast) for efficiency. Models in Regime A are trained for 60,000 optimization steps; Regime B runs use 20,000–30,000 steps depending on the sweep configuration.

5.3 Evaluation Protocol

We evaluate on the held-out validation split every 1,000 steps (or every 500 steps for shorter runs), saving the checkpoint with the lowest validation MSE (`ckpt_best.pt`). The final scaling analysis uses the best validation MSE achieved during training for each run, which we denote L^*. Each model size in Regime A is trained with three random seeds (7, 8, 9) to characterize run-to-run variance; Regime B coverage varies by model size.

Table 2: Canonical Regime A summary ($V = 512$, 200k cells, 60k steps, 3 seeds per size). Mean $\pm$ standard deviation is reported across seeds for both validation and held-out test MSE.

Size	Params	Val. MSE	Test MSE
XXS	533	2.101 ± 0.032	2.110 ± 0.033
TINY	9,953	1.600 ± 0.010	1.617 ± 0.007
XS	132,993	1.515 ± 0.011	1.533 ± 0.007
S	859,137	1.482 ± 0.008	1.500 ± 0.007
M	19,178,497	1.462 ± 0.003	1.479 ± 0.004
L	100,510,801	1.477 ± 0.009	1.494 ± 0.011

6 Scaling Law Formulation

Following Kaplan et al. [1], we model the relationship between validation MSE and parameter count P as a power law with an additive offset:

$$L(P) = a \cdot P^{-\alpha} + c, \tag{3}$$

where $\alpha > 0$ is the *scaling exponent* governing the rate of improvement, $a > 0$ is a multiplicative constant, and $c \geq 0$ represents the *irreducible loss floor*—the error that cannot be reduced by increasing model capacity, attributable to intrinsic data noise, biological stochasticity, or information lost during preprocessing.

We fit Equation (3) by grid search over candidate values of c in $[0, 0.99 \cdot \min_i L_i]$, and for each candidate perform ordinary least squares on the linearized form $\log(L - c) = \log a - \alpha \log P$. The fit with the highest R^2 on the log-scale residuals is selected. In the revised manuscript, we report main-text fits only for standardized sweeps; exploratory mismatched runs are described qualitatively rather than pooled into a single regression.

7 Results

7.1 Overview of Standardized and Exploratory Runs

Our main quantitative evidence now focuses on the standardized Regime A sweep: 18 runs (6 model sizes $\times$ 3 seeds), all trained for 60,000 steps with identical optimization hyperparameters on the same $V = 512$, $D = 200$k dataset. Three earlier 30k-step Regime A runs are retained only as exploratory pilots and are excluded from all main-text regressions and summary tables. We retain the previously reported $V = 1{,}024$ runs only as an exploratory inventory, because those runs vary in dataset size and optimization budget.

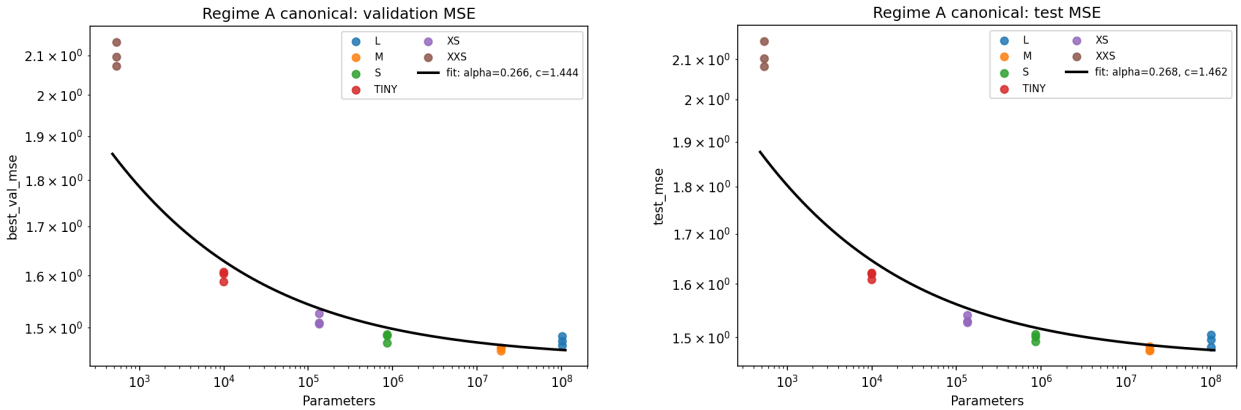
In the canonical $V = 512$ sweep, MSE decreases steadily from XXS through M on both validation and held-out test data. The L model does not improve further and is slightly worse than M on average, indicating that the loss curve has already begun to flatten within the explored parameter range.

7.2 Canonical Regime A Scaling

Table 3 reports the power-law fit parameters for the canonical Regime A sweep on both validation and held-out test MSE, and Figure 1 shows the corresponding log-log fits.

Table 3: Power-law fits for the canonical Regime A sweep ($V = 512$, 200k cells, 18 runs). Reporting both validation and held-out test fits confirms that the qualitative conclusion does not depend on checkpoint selection by validation alone.

Split	n	α	a	c	R^2
Validation	18	0.266	2.153	1.444	0.858
Test	18	0.268	2.168	1.462	0.859



(a) Validation MSE.

(b) Held-out test MSE.

Figure 1: Canonical Regime A scaling on log-log axes. Each point is one training run (best checkpoint per seed). The held-out test fit closely matches the validation fit, with $\alpha \approx 0.267$ and $R^2 \approx 0.859$ on both splits.

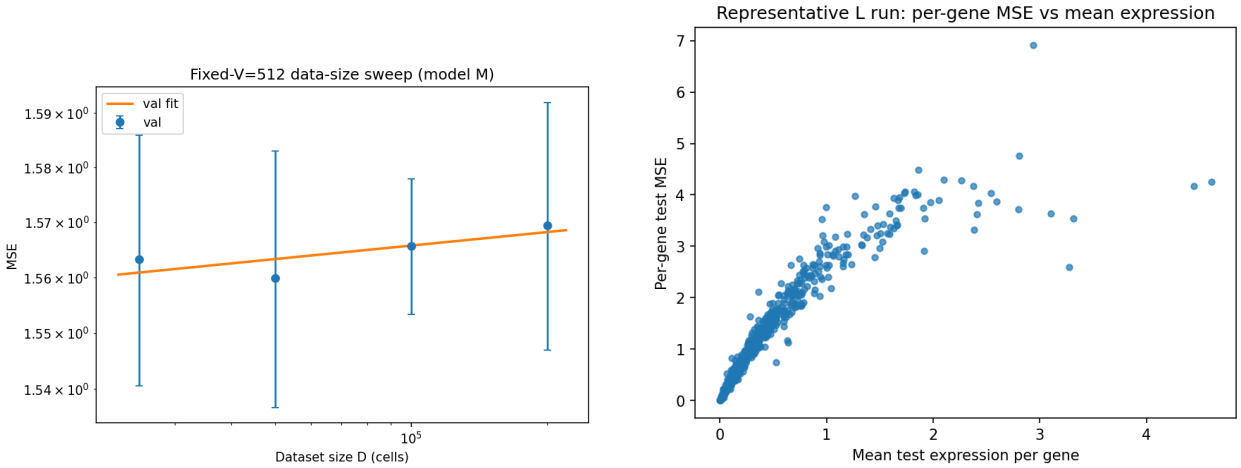
The standardized $V = 512$ regime with 200,000 cells exhibits a clear scaling relationship. On validation data, the fitted exponent is $\alpha = 0.266$ with $R^2 = 0.858$. On held-out test data, the fit is nearly identical ($\alpha = 0.268$, $R^2 = 0.859$), showing that the trend is not merely an artifact of selecting checkpoints by validation performance. In the canonical fit, each tenfold increase in parameter count reduces the excess loss $(L - c)$ by a factor of approximately $10^{0.266} \approx 1.84$. The irreducible floor is also stable across splits: $c \approx 1.44$ on validation and $c \approx 1.46$ on test.

The observation that L-size models ($\sim$101M parameters) do not improve over M-size models ($\sim$19M parameters) is consistent with the fitted curve, which has largely flattened by $P \approx 10^7$. This provides practical guidance: for this dataset and task, models beyond the tens-of-millions scale offer diminishing returns.

7.3 Exploratory $V = 1{,}024$ Inventory

We no longer report a single pooled scaling-law fit for the originally submitted $V = 1{,}024$ runs in the main text. Those runs are heterogeneous: the retained exploratory inventory spans both $D = 10$k and $D = 50$k datasets, total training budgets from 20k to 60k steps, and only partial size coverage. Accordingly, they cannot support a clean causal statement about "data scarcity" or a standardized negative result.

The useful conclusion is therefore narrower than in the original draft. The exploratory $V = 1{,}024$ runs are consistent with weaker or less stable scaling under those settings, but they do not isolate whether the difference is driven by dataset size, vocabulary size, optimization budget, or their



(a) Fixed-V data-size sweep at model size M.

(b) Per-gene test MSE vs. mean expression.

Figure 2: Follow-up analyses from existing $V = 512$ runs. Left: at fixed 60k steps, the M-model data-size sweep is nearly flat in validation MSE ($\beta \approx -0.002$, $R^2 = 0.61$), indicating that a clean $L(D)$ law still requires matched optimization budgets. Right: residual error is strongly gene-dependent in a representative held-out L run, with correlation 0.892 between mean expression and per-gene MSE.

interaction. Matched V/D sweeps remain necessary for that question.

7.4 Matched-V Follow-up Analyses

To partially address the confound in the original two-regime comparison, we additionally analyzed matched-$V = 512$ follow-ups from existing runs. First, we evaluated a fixed-model data-size sweep using the M model at $D \in \{25\text{k}, 50\text{k}, 100\text{k}, 200\text{k}\}$ with 3 seeds per point and the same 60k-step training budget. Second, we computed a per-gene held-out residual diagnostic on a representative L run to test the homoscedasticity assumption underlying the MSE-to-entropy conversion.

The fixed-V data-size sweep yields a nearly flat validation fit, $\beta = -0.0023$ with $R^2 = 0.611$, rather than a clean monotonic decrease in MSE with larger D. This does not contradict the importance of data; rather, it shows that at fixed total steps, larger datasets are seen for fewer effective epochs, so the current sweep is not yet a definitive $L(D)$ estimate. It nevertheless helps deconfound the main narrative: the original Regime A vs. Regime B contrast should not be read as simple evidence that "more cells alone" explains the difference.

The residual diagnostic also shows that the homoscedastic Gaussian approximation is imperfect. In the representative held-out L run, the across-gene coefficient of variation of per-gene test MSE is 0.730, the ratio between the 90th and 10th percentiles is 10.1, and the correlation between mean expression and per-gene MSE is 0.892. We therefore treat the entropy-style conversion from MSE to bits as a heuristic estimate of irreducible uncertainty under the task and preprocessing pipeline, not as a justified gene-wise generative model.

Finally, we regenerated a supplementary Section 6.3-style $V = 512$ analysis from the cleaned run manifest with phase-specific filtering, combining the fixed-regime sweep, the matched-V data sweep, and the available compute-sweep families. In that supplementary analysis, the newer $V = 512$ follow-up runs contribute directly logged Gaussian NLL values, so the compute-frontier study is not limited to post hoc MSE conversion alone. We do not elevate those history-based compute-frontier fits to a main-text claim, because they still fall short of a fully matched $V/D/C$ design, but they are

consistent with the same low-2-bit range suggested by the canonical floor fit.

7.5 Comparison with Language Model Scaling

The scaling exponent $\alpha \approx 0.266$–0.268 observed in the canonical Regime A sweep is substantially larger than the $\alpha_P \approx 0.076$ reported by Kaplan et al. [1] for autoregressive language models, or the $\alpha_P \approx 0.34$ reported by Hoffmann et al. [3] after correcting for compute-optimal training. Direct comparison is complicated by several factors: we use MSE rather than cross-entropy (Appendix A), our masking rate (15%) is much lower than typical language model sequence lengths, and the structure of gene expression data differs fundamentally from natural language. Nevertheless, the qualitative finding—that loss follows a power law with diminishing returns at large scale—is consistent across domains.

The contrast between our canonical and exploratory settings echoes the broader observation of Kaplan et al. [1] that different empirical regimes can coexist for scaling with parameters, data, and compute. In our setting, however, the current matrix is too confounded to claim a direct analogue of Section 6.3 on the basis of the cross-regime comparison alone.

7.6 Preliminary Entropy Estimate for Regime A

A central motivation for fitting scaling laws with an additive offset c is that the irreducible floor can, under appropriate assumptions, be converted to an estimate of the *intrinsic entropy* of the data-generating process—the information content per prediction that no model can recover. While a full entropy estimate requires matched data/compute sweeps and a likelihood-based objective (Section 9), we present a preliminary estimate for Regime A using Gaussian approximations to the fitted floor values.

7.6.1 Gaussian NLL Derivation from MSE

Our canonical training runs record MSE as the primary loss. To obtain a Gaussian negative log-likelihood (NLL), we derive it post hoc under the assumption that the residual errors at masked positions are approximately Gaussian with variance equal to the MSE and, crucially, that this residual variance can be treated as homoscedastic across genes. Specifically, for a Gaussian with variance σ^2, the NLL per position is:

$$\mathcal{L}_{\text{NLL}} = \tfrac{1}{2}\ln(2\pi\sigma^2) + \tfrac{1}{2}, \tag{4}$$

and the differential entropy (in bits) is:

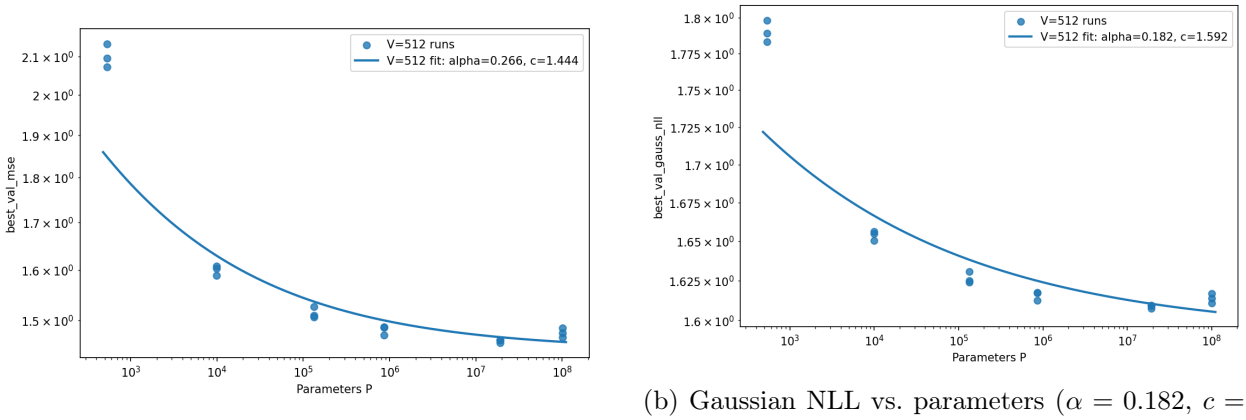
$$h = \tfrac{1}{2}\log_2(2\pi e\,\sigma^2). \tag{5}$$

Using a standardized set of 18 canonical Regime A runs (6 sizes $\times$ 3 seeds, all trained for 60,000 steps with identical hyperparameters), we refit the scaling law for both the MSE and the derived Gaussian NLL metrics. As a limited direct check on the conversion, we also reran XXS and TINY under the same $V = 512$, $D = 200$k, 60k-step protocol with validation Gaussian NLL logged directly; those supplementary results are used only as a calibration check, not as a replacement for the six-size canonical fit. Table 4 reports the results.

Figure 3 shows the scaling curves for both metrics on the canonical run set.

Table 4: Scaling-law fits for Regime A ($V = 512$, 200k cells) using the 18 canonical runs, with entropy floor estimates. The NLL values are derived from MSE under a Gaussian assumption (Equation 4), so the two estimates are not fully independent; Figure 2 shows that residual variance is in fact gene-dependent.

Metric	n	α	a	c	R^2	**Entropy floor (bits/masked pos.)**
MSE	18	0.266	2.153	1.444	0.858	2.312
Gauss. NLL	18	0.182	0.401	1.592	0.838	2.296



(a) MSE vs. parameters ($\alpha = 0.266$, $c = 1.444$).

(b) Gaussian NLL vs. parameters ($\alpha = 0.182$, $c = 1.592$).

Figure 3: Canonical Regime A scaling fits (18 runs, 6 sizes $\times$ 3 seeds, all 60k steps). Left: MSE metric. Right: derived Gaussian NLL metric. Under the Gaussian homoscedastic approximation, both correspond to about 2.30 bits of irreducible uncertainty remaining per masked position.

7.6.2 Entropy Estimate

The two derivations yield closely agreeing estimates:

- **From MSE floor** ($c_{\text{MSE}} = 1.444$): interpreting c as the irreducible variance σ^2 and applying Equation (5) gives $h = \frac{1}{2}\log_2(2\pi e \cdot 1.444) = 2.312$ bits per masked position.
- **From NLL floor** ($c_{\text{NLL}} = 1.592$ nats): converting to bits via $h = c_{\text{NLL}}/\ln 2 = 2.296$ bits per masked position.

Both approaches converge on an estimate of approximately **2.30 bits of irreducible uncertainty remaining per masked gene position** for the $V = 512$ regime, but only under the stated Gaussian and homoscedasticity assumptions.

7.6.3 Caveats

This estimate is preliminary and subject to several important caveats. First, the Gaussian NLL values in the canonical sweep are *derived from* the MSE values (not independently logged during training), so the two estimates in Table 4 are not fully independent—their agreement primarily reflects the self-consistency of the Gaussian assumption rather than an independent validation. The new direct-logging check partially addresses this only at small scale: across three-seed reruns, the mean validation Gaussian NLL differs from the MSE-derived value by 0.022 nats for XXS (1.768

direct vs. 1.790 derived) and 0.020 nats for TINY (1.634 direct vs. 1.654 derived). This supports the local calibration of Equation (7) for those two small models, but it does not replace the six-size canonical table, because the available directly logged XS/S/M/L histories belong to a later follow-up sweep rather than to the original canonical family.

Second, the entropy floor depends on the preprocessing pipeline. The $\log(1+x)$ transform and library-size normalization change the scale and distribution of gene expression values; the 2.30-bit figure applies specifically to masked positions in $\log(1+x)$-transformed, library-size-normalized expression over 512 HVGs. Changing the vocabulary size, normalization target, or transformation would alter the floor.

Third, a complete entropy estimate requires not only the $L(P)$ scaling law (loss vs. parameters) but also the $L(D)$ law (loss vs. dataset size) and ideally the intersection $L(C_{\min})$ of the compute-optimal frontier, following the methodology of Hoffmann et al. [3]. We now include a matched-$V = 512$ follow-up data-size sweep, but at a fixed 60k-step budget it is nearly flat in MSE and therefore does not yet provide a clean $L(D)$ law; a definitive estimate still requires matched optimization budgets across D.

Fourth, the homoscedasticity assumption is only approximate in our data. A representative held-out L run shows substantial cross-gene heterogeneity in residual error (coefficient of variation 0.730, $q_{90}/q_{10} = 10.1$, correlation 0.892 between mean expression and per-gene MSE; Figure 2). This does not invalidate the floor-fit itself, but it means the conversion from MSE to bits should be interpreted as a task-level approximation rather than a justified per-gene likelihood model.

Despite these limitations, the convergence of the two derivations on ∼2.30 bits provides a useful first benchmark: under the stated approximation, it suggests that roughly 2.3 bits of *irreducible uncertainty remain* per masked gene among the top 512 HVGs, given this data source, preprocessing pipeline, and masking task.

8 Discussion

8.1 Implications for Single-Cell Foundation Models

Our results carry several implications for the design and training of single-cell foundation models.

Data availability likely matters, but the original comparison is not causal. The sharp contrast between Regimes A and B is consistent with the hypothesis that data quantity relative to model capacity matters strongly for masked-reconstruction pretraining. However, because the original comparison changes both D and V simultaneously and Regime B is heterogeneous in training budget, it should not be read as direct evidence that data scarcity alone is the binding constraint. The more defensible conclusion is narrower: under the canonical $V = 512$, $D = 200$k setup, larger models help; under the originally reported exploratory $V = 1{,}024$ settings, they do not provide a similarly clean gain. Moreover, our matched-$V = 512$ fixed-step data sweep is nearly flat in MSE, so the current evidence does not support a simple monotonic "more cells ⇒ lower loss" story without also controlling optimization budget.

Diminishing returns beyond ∼20M parameters. Within Regime A, the loss curve flattens around 10^7 parameters. While this specific threshold depends on the dataset, gene vocabulary, and preprocessing choices, it suggests that for datasets of ∼200,000 cells, models in the tens of millions of parameters are sufficient to capture the learnable structure. This is a useful calibration point for practitioners deciding whether to invest in training larger models.

The irreducible floor as a biological signal. The estimated $c \approx 1.44$ (MSE) for Regime A, corresponding under the Gaussian homoscedastic approximation to approximately 2.30 bits of irreducible uncertainty remaining per masked gene position (Section 7.6), reflects the combined effect of biological noise, technical noise, and information loss from dimensionality reduction. Disentangling these contributions is an important direction for future work: if the floor is dominated by technical noise, it may be reducible through improved experimental protocols; if it reflects intrinsic biological stochasticity, it represents a fundamental limit on expression prediction. The 2.30-bit figure should therefore be interpreted as an approximate, task-specific quantity of irreducible uncertainty under this preprocessing pipeline, masking objective, and gene vocabulary, rather than as a universal constant of transcriptomics.

8.2 The Data-vs-Parameter Ratio

A plausible interpretation of our two regimes is that scaling behaviour depends on the interaction between data, parameter count, and optimization budget. In Regime A, even the L model ($\sim$101M parameters) is trained on 200,000 cells $\times$ 512 genes $\approx 10^8$ values, giving a ratio of $\sim$1 datum per parameter. In Regime B, the XL model ($\sim$341M parameters) sees only 10,000 $\times$ 1,024 $\approx 10^7$ values, a ratio of $\sim$0.03 data per parameter. However, the matched-$V = 512$ follow-up shows that holding steps fixed while varying D is not enough to produce a clean data law, because the effective number of epochs changes with dataset size. The transition from effective to ineffective scaling therefore cannot be localized from the current matrix alone; isolating it requires matched V/D sweeps with standardized optimization budgets.

We note that the effective data budget is further multiplied by the number of training steps and the mask rate: each cell is seen multiple times with different random masks, effectively augmenting the dataset. A full accounting of *tokens seen* (steps $\times$ effective batch size $\times V$) would enable a cleaner analysis of data-vs-compute scaling, which we leave to future work.

8.3 Comparison to Downstream Evaluations

Prior single-cell foundation models are typically evaluated on downstream tasks (cell-type classification, batch correction, perturbation response prediction) rather than pretraining loss. Our work is complementary: we focus on the pretraining loss itself, which is the quantity most directly comparable across model sizes and most amenable to scaling-law analysis. A natural extension would be to study whether downstream task performance also follows predictable scaling curves, as has been observed in some NLP settings [23, 24].

9 Limitations and Future Work

Confounded regime comparison. The two regimes differ in both gene vocabulary size (V) and dataset size (D) simultaneously. This confounding prevents us from isolating the effect of each factor. Future work should construct a matrix of experiments varying V and D independently, *e.g.* by building $V = 512$ and $V = 1{,}024$ datasets at matched cell counts (10k, 50k, 200k each).

MSE as loss metric and derived entropy. We use MSE as both the training loss and the scaling-law target. While we provide a preliminary entropy estimate of $\sim$2.30 bits per masked position for Regime A (Section 7.6), this estimate relies on a Gaussian assumption applied post hoc to MSE values, rather than a natively trained information-theoretic loss. The Gaussian NLL values in the original canonical sweep are derived from MSE, not independently logged, so the two

entropy derivations are not fully independent. To partially test this approximation, we completed direct-logging XXS/TINY reruns under the same canonical $V = 512$, $D = 200$k, 60k-step setup; the mean validation Gaussian NLL is within about 0.02 nats of the value derived from best validation MSE for both sizes. The cleaned supplementary Section 6.3-style analyses reported here also use directly logged Gaussian NLL where available in the newer $V = 512$ follow-up runs. However, the canonical scaling table itself remains MSE-based, because we do not yet have a fully matched six-size direct-NLL rerun of the original canonical family, and the homoscedastic Gaussian conversion remains only approximate. To obtain a rigorous transcriptomic entropy estimate, future work should train with a parametric Gaussian NLL that predicts both μ and gene-specific σ^2, or with a discretized cross-entropy over binned expression levels. Additionally, the full $L(D)$ and $L(C_{\min})$ intersection experiment described by Hoffmann et al. [3] is needed to disentangle data-limited from model-limited floors.

Limited model size range. While our seven presets span five orders of magnitude, the "useful" range for Regime A is narrower: the XXS model (533 parameters) is too small to be informative about neural scaling behaviour, and models beyond L show no improvement due to the data constraint. A denser sweep in the 10^5–10^7 range, with more seeds, would yield tighter confidence intervals on α and c.

Training compute as a scaling axis. We have studied scaling with respect to parameters only. A complete picture requires also fitting $L(C)$ as a function of compute C (measured in FLOPs or tokens processed), and $L(D)$ as a function of dataset size. Our runs do record tokens processed (steps $\times$ effective batch $\times V$), and the supplementary $V = 512$ follow-up analyses now provide initial history-based fits for $L(D)$ and a compute-frontier proxy. All such supplementary summaries cited in this revision were regenerated from the current $V = 512$ run manifest with phase-specific filtering, rather than copied from earlier mixed aggregate folders. However, these follow-ups do not yet replace a fully matched $V/D/C$ experimental matrix, which remains necessary for a definitive Section 6.3-style entropy estimate.

Heterogeneous training configurations. The Regime B runs used varying training durations and batch sizes, introducing noise that obscures potential scaling trends. Future sweeps should use strictly standardized hyperparameters across all model sizes within a regime.

Generalization to other tissues, species, and modalities. Our data are drawn from a single Census subset. It remains to be seen whether the observed scaling exponents and floors generalize across tissue types, developmental stages, species, or multi-omic modalities (ATAC-seq, spatial transcriptomics).

10 Broader Impact

This work is primarily methodological, but its conclusions inherit biases from the underlying single-cell compendium and from preprocessing choices such as HVG selection, library-size normalization, and the $\log(1 + x)$ transform. Dataset composition can over-represent certain tissues, platforms, and donor populations, so any observed "irreducible" floor may partly reflect those biases rather than purely biological uncertainty. Similarly, the entropy-style quantity discussed in Section 7.6 is task-dependent: it is specific to masked reconstruction over a chosen gene vocabulary after

a particular preprocessing pipeline, and it should not be interpreted as a universal constant of transcriptomics.

11 Conclusion

We have presented the first systematic study of neural scaling laws for masked-reconstruction transformers on single-cell RNA-seq data. Our principal finding is that scaling behaviour analogous to that observed in language modelling does emerge in single-cell transcriptomics, but only when sufficient data are available relative to model capacity. In the canonical $V = 512$, $D = 200$k regime, validation MSE follows a power law in parameter count with an identifiable irreducible floor, and held-out test evaluation follows the same qualitative pattern. By contrast, the originally reported $V = 1{,}024$, $D = 10$k comparison does not justify a strong causal claim because it is confounded and heterogeneous. A preliminary conversion of the asymptotic loss floor to information-theoretic units yields an estimate of approximately 2.30 bits of irreducible uncertainty per masked gene position under the stated Gaussian approximation.

These results establish a quantitative framework for reasoning about the scaling behaviour of single-cell foundation models. They suggest that the field's current emphasis on increasing model size should be balanced by equivalent investment in curating larger, more diverse training corpora and in running matched sweeps that disentangle dataset size, vocabulary size, and compute. With the addition of natively trained information-theoretic loss functions, matched experimental conditions, and comprehensive compute accounting, this framework can be refined to produce rigorous estimates of the intrinsic entropy of the transcriptome—a quantity of fundamental biological and information-theoretic interest.

Acknowledgements

We thank the CELLxGENE team at the Chan Zuckerberg Initiative for maintaining the Census data resource, and the developers of scvi-tools, Scanpy, and PyTorch for the software infrastructure underlying this work. Compute was provided by consumer GPU hardware.

References

[1] J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, and D. Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

[2] J. Hestness, S. Narang, N. Ardalani, G. Diamos, H. Jun, H. Kianinejad, M. M. A. Patwary, Y. Yang, and Y. Zhou. Deep learning scaling is predictable, empirically. *arXiv preprint arXiv:1712.00409*, 2017.

[3] 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. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[4] X. Zhai, A. Kolesnikov, N. Houlsby, and L. Beyer. Scaling vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[5] M. Cherti, R. Beaumont, R. Wightman, M. Wortsman, G. Ilharco, C. Gordon, C. Schuhmann, L. Schmidt, and J. Jitsev. Reproducible scaling laws for contrastive language-image learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[6] A. Clark, D. de Las Casas, A. Guy, A. Sherburn, D. Sherburn, and S. Sherburn. Unified scaling laws for routed language models. In *International Conference on Machine Learning (ICML)*, 2022.

[7] N. Muennighoff, A. Rush, B. Barak, T. Le Scao, A. Piktus, N. Tazi, S. Pyysalo, T. Wolf, and C. Raffel. Scaling data-constrained language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

[8] A. Regev, S. A. Teichmann, E. S. Lander, I. Amit, C. Benoist, D. Birney, C. Bodenmiller, P. Campbell, P. Carninci, M. Clatworthy, et al. The Human Cell Atlas. *eLife*, 6:e27041, 2017.

[9] CZI Single-Cell Biology. CZ CELLxGENE Discover: A single-cell data platform for scalable exploration, analysis, and modeling of aggregated data. *bioRxiv*, 2023. doi:10.1101/2023.10.30.563174.

[10] 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, 2018.

[11] A. Gayoso, R. Lopez, G. Xing, P. Boyeau, V. Valiber Pine, M. Lotfollahi, M. Schiebinger, and N. Yosef. A Python library for probabilistic analysis of single-cell omics data. *Nature Biotechnology*, 40(2):163–166, 2022.

[12] H. Cui, C. Wang, H. Maan, K. Pang, F. Luo, N. Doshi, and B. Wang. scGPT: Toward building a foundation model for single-cell multi-omics using generative AI. *Nature Methods*, 21(8):1470–1480, 2024.

[13] C. V. Theodoris, L. Xiao, A. Chopra, M. D. Chaffin, Z. R. Al Sayed, M. C. Hill, H. Manberola, E. Sayed, J. Karber, E. Ellinor, and P. T. Ellinor. Transfer learning enables predictions in network biology. *Nature*, 618:616–624, 2023.

[14] F. Yang, W. Wang, F. Wang, Y. Fang, D. Tang, J. Huang, H. Lu, and J. Yao. scBERT as a large-scale pretrained deep language model for cell type annotation of single-cell RNA-seq data. *Nature Machine Intelligence*, 4:852–866, 2022.

[15] 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.

[16] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, 2019.

[17] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[18] T. Stuart, A. Butler, P. Hoffman, C. Hafemeister, E. Papalexi, W. M. Mauck, Y. Hao, M. Stoeckius, P. Smibert, and R. Satija. Comprehensive integration of single-cell data. *Cell*, 177(7):1888–1902, 2019.

[19] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[20] D. Hendrycks and K. Gimpel. Gaussian error linear units (GELUs). *arXiv preprint arXiv:1606.08415*, 2016.

[21] R. Xiong, Y. Yang, D. He, K. Zheng, S. Zheng, C. Xing, H. Zhang, Y. Lan, L. Wang, and T.-Y. Liu. On layer normalization in the transformer architecture. In *International Conference on Machine Learning (ICML)*, 2020.

[22] I. Loshchilov and F. Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, 2019.

[23] J. Wei, Y. Tay, R. Bommasani, C. Raffel, B. Zoph, S. Borgeaud, D. Yogatama, M. Bosma, D. Zhou, D. Metzler, E. H. Chi, T. Hashimoto, O. Vinyals, P. Liang, J. Dean, and W. Fedus. Emergent abilities of large language models. *Transactions on Machine Learning Research*, 2022.

[24] B. Isik, N. Ponomareva, H. Hazimeh, D. Paparas, S. Vassilvitskii, and S. Kale. Scaling laws for downstream task performance of large language models. In *Findings of ACL*, 2024.

A On the Choice of Loss Metric

Mean squared error is the most common regression loss for continuous expression values, but it is not directly comparable to the cross-entropy losses used in language model scaling laws. Ideally, one would train with a Gaussian negative log-likelihood where the model predicts both a mean μ_g and a variance σ_g^2:

$$\mathcal{L}_{\text{NLL}} = \frac{1}{|\mathcal{M}|} \sum_{g \in \mathcal{M}} \left[\frac{(x_g - \mu_g)^2}{2\sigma_g^2} + \log \sigma_g + \frac{1}{2} \log(2\pi) \right]. \tag{6}$$

This formulation measures loss in nats (convertible to bits by dividing by $\ln 2$) and would allow the asymptotic floor c to be directly interpreted as an entropy estimate.

In the current study, we derive Gaussian NLL values post hoc from MSE by assuming homoscedastic Gaussian residuals with variance $\sigma^2 = \text{MSE}$:

$$\hat{\mathcal{L}}_{\text{NLL}} = \tfrac{1}{2} \ln(2\pi \cdot \text{MSE}) + \tfrac{1}{2}. \tag{7}$$

This derivation is exact only if the prediction errors are truly $\mathcal{N}(0, \text{MSE})$ with a variance that can be treated as constant across genes, but it conflates aleatoric and epistemic uncertainty and cannot capture heteroscedastic noise across genes. In our 18 canonical Regime A runs, all Gaussian NLL values carry the provenance label `derived_from_best_mse` in the analysis pipeline, confirming that they are not independently measured. As a partial empirical check, we reran XXS and TINY under the same canonical $V = 512$, $D = 200$k, 60k-step configuration with validation Gaussian NLL logged directly. Across three seeds, the directly logged validation Gaussian NLL is 1.768 vs. 1.790 derived for XXS, and 1.634 vs. 1.654 derived for TINY, *i.e.* within about 0.02 nats in both cases. These small-model reruns support the local numerical calibration of the conversion, but they do not constitute a full six-size direct-NLL replacement for the canonical table because the available larger direct-NLL histories come from a later, non-identical follow-up sweep. The revision also adds a held-out per-gene residual diagnostic; in a representative L run, the per-gene test-MSE coefficient

of variation is 0.730 and the q_{90}/q_{10} ratio is 10.1, reinforcing that the homoscedastic conversion is only approximate.

The entropy conversion from each floor then proceeds as follows:

$$h_{\mathrm{NLL}} = c_{\mathrm{NLL}}/\ln 2 = 1.592/0.693 = 2.296 \text{ bits}, \tag{8}$$

$$h_{\mathrm{MSE}} = \tfrac{1}{2}\log_2(2\pi e \cdot c_{\mathrm{MSE}}) = \tfrac{1}{2}\log_2(2\pi e \cdot 1.444) = 2.312 \text{ bits}. \tag{9}$$

The close agreement ($\Delta = 0.016$ bits) reflects the self-consistency of the Gaussian assumption rather than independent validation. A per-gene residual diagnostic therefore matters for interpretation: if residual variance differs substantially across genes, the conversion should be read as an approximation to task-specific irreducible uncertainty rather than as a justified generative model. A natively trained NLL model—where the network outputs both μ_g and $\log \sigma_g^2$—would provide a genuinely independent estimate and is the natural next step.

B Detailed Run Configuration

All runs use the following shared configuration unless otherwise noted:

- **Mask rate**: 15%
- **Optimizer**: AdamW with $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$
- **Weight decay**: 0.01
- **Gradient clipping**: max norm 1.0
- **Mixed precision**: float16 autocast on GPU
- **Activation**: GELU
- **Normalization**: Pre-LN (layer norm before attention and FFN)
- **Pooling**: mean pooling over gene tokens
- **Prediction head**: single linear layer
- **Initialization**: Xavier uniform for linear weights; normal ($\sigma = 0.02$) for embeddings

Regime A (GPU) runs use physical batch size 4, gradient accumulation 8 (effective batch 32), learning rate 3.125×10^{-5}, and 60,000 steps. Three additional 30k-step Regime A pilot runs exist in the repository, but they are treated as exploratory sanity checks and excluded from the main-text standardized sweep. The matched-$V = 512$ follow-up data sweep keeps the same optimization settings while varying dataset size across 25k, 50k, 100k, and 200k cells with model size M. The exploratory $V = 1{,}024$ inventory mixes datasets of 10k and 50k cells and total budgets from 20,000 to 60,000 steps, so we do not pool those runs into a main-text regression.

C Reproducibility

All code, data-building scripts, training loops, and analysis pipelines are available in the accompanying repository. Datasets are constructed deterministically from the CELLxGENE Census with fixed random seeds (seed 42 for data splits, seeds 7/8/9 for model training). The full set of training run outputs—checkpoints, history logs, and metadata—is preserved in the `runs/` directory. For the supplementary $V = 512$ follow-up analyses in this revision, all aggregate summaries were recomputed from the current run manifest with explicit phase filtering before citation.