Cross-Modality Representation Interpretation for Single-Cell Language Models

Anonymous authors
Paper under double-blind review

Abstract

Single-cell foundation models continue to improve on standard benchmarks, but it remains unclear whether their internal representations encode biological knowledge or merely capture subtle expression patterns correlated with benchmark labels. We investigate this question in Cell-to-Sentence-Scale models, which represent cells as gene-name sentences and thus provide a natural setting for examining whether language-style training helps models capture biological knowledge. Using two human peripheral blood mononuclear cell datasets, we train sparse autoencoders (SAEs) on gene-level residual-stream activations from several intermediate layers, and evaluate the learned feature dictionaries along three axes: representation quality, downstream utility, and biological interpretability. These learned features are broadly utilized, predominantly selective, and nonredundant, with these properties persisting under distributional shifts in the input data. Replacing gene embeddings with their SAE reconstructions largely preserves downstream task performance, suggesting that the information discarded by the SAEs has limited relevance to the tasks evaluated. Gene Ontology enrichment analysis further reveals that feature semantics sharpen with layer depth, progressing from generic immune to lineage-specific programs. Steering with interpretable biological concepts shifts cell-type predictions in the intended directions, providing further evidence that the foundation model learns biologically meaningful representations. This control varies across layers, while SAE-reconstructed representations preserve the model's responsiveness to feature-level perturbations. Together, these findings provide a practical framework for interpreting cross-modal single-cell foundation models and diagnosing their learned representations.

1 Introduction

Single-cell foundation models (scFMs) have emerged as an increasingly important direction in single-cell genomics, driven by the rapid growth of large-scale single-cell atlases and the success of self-supervised pretraining. These models aim to learn general-purpose representations of genes and cells that can support downstream tasks such as cell-type annotation and perturbation prediction (Theodoris et al., 2023; Cui et al., 2024; Hao et al., 2024). However, despite growing efforts to scale pretraining datasets and model parameters, relatively little attention has been paid to whether scFMs acquire biologically meaningful internal representations. In particular, it remains unclear whether these models encode biological knowledge, such as pathways and regulatory relationships, or whether their apparent performance is largely driven by memorizing training data and capturing benchmark-specific correlations. Addressing this question is important not only for model interpretability, but also for developing the next generation of more powerful scFMs.

Many scFMs adopt the Transformer architecture, partly inspired by its broad success in large language models (LLMs). However, tokenization and embedding strategies for single-cell data remain open and nontrivial challenges. Unlike natural language, single-cell RNA-sequencing (scRNA-seq) data are high-dimensional tabular data, sparse, noisy, and unordered: genes do not form a canonical sequence. Conventional scFM representations therefore typically combine multiple components, including gene identity embeddings,

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

embeddings or discretized representations of expression levels, and, in some cases, positional encodings based on gene expression ranks. More recently, Cell-to-Sentence-Scale (C2S) models have represented cells as sequences of gene names ranked by expression and processed these sequences using a language-model backbone (Rizvi et al., 2026). These models have demonstrated strong performance across multiple benchmarks, suggesting that language-style representations help capture patterns useful for downstream tasks. However, benchmark performance alone does not establish whether these representations capture generalizable biological knowledge.

C2S models provide a setting for investigating the relationship between biological input data and language-based representations. This motivates two distinct questions: do their internal representations contain features associated with recognizable biological concepts, and do interventions on those features alter model behavior? Sparse autoencoders (SAEs) offer a way to investigate these questions by decomposing foundation model internal dense activations into sparse feature dictionaries (Cunningham et al., 2023; Gao et al., 2025). Prior work has applied SAEs to protein language models and expression-based single-cell models to identify biological associations and, in some settings, steer model outputs (Simon & Zou, 2025; Schuster, 2025; Claye et al., 2025; Pedrocchi et al., 2025). Building on these studies, we examine how biological feature organization and intervention sensitivity relate to one another across layers of a single-cell language model. An extended discussion of related work is provided in Appendix A.

We study C2S Gemma-2-2B model across three representative immune cell types by training sparse autoencoders (SAEs) on gene-level residual-stream activations and linking the learned features to genes, cell types, and Gene Ontology (GO) terms. We first test whether SAE reconstructions preserve cell-type annotation performance, then characterize how features are organized across model layers, and finally evaluate the effects of feature interventions through steering. Our main contributions are[1]:

- To our knowledge, we present one of the first systematic interpretability studies of a cross-modal single-cell foundation model, introducing a framework that connects single cell language model representations to biological knowledge.
- We show that SAE-reconstructed embeddings can preserve downstream task performance, supporting the use of SAEs as reliable embedding translators that convert black-box embeddings into more interpretable representations.
- We connect SAE features to known biological concepts, analyze their organization across three layers, and test whether these features influence model behavior in biologically coherent ways through steering experiments.

2 Interpretation Framework and Experimental Setup

We summarize our interpretation pipeline in Figure 1. We train a separate SAE on gene activations extracted from each examined layer of the frozen C2S model, then use the learned feature dictionaries for reconstruction, biological annotation, and intervention.

2.1 Models, Data, and Activation Extraction

We study the representative cross-modality scFM C2S Gemma-2-2B, with SAEs trained at layers $\ell \in \{10, 15, 20\}$. These middle-to-late layers enable comparisons of feature organization across model depth, motivated by prior SAE studies (Gallifant et al., 2025; Sun et al., 2025). Following the C2S preprocessing convention, counts are normalized per cell and log-transformed, and the highest-ranked genes are used to construct a cell sentence. Because the SentencePiece tokenizer can split gene names into multiple tokens (Kudo & Richardson, 2018), we represent each gene using the residual-stream activation at its final token.

We analyze B cells, CD14 monocytes, and CD56 natural killer (NK) cells, spanning major lymphoid and myeloid populations. SAE training and validation use only `purified-PBMC` (Zheng et al., 2017), while `PBMC-4k` and `PBMC-8k` (10x Genomics, 2017a;b),

[1]Our code and results are available in an anonymous repository; SAE checkpoints will be released after the review period.

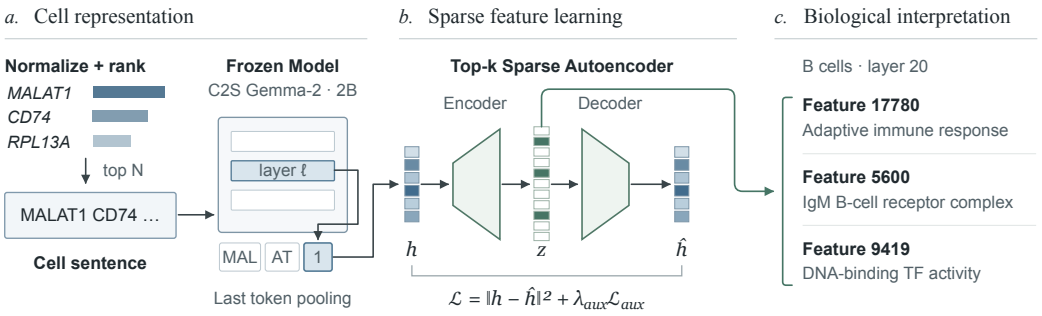


Figure 1: **Framework for interpreting C2S representations.** A separate SAE is trained at each examined layer, and the resulting SAE features are analyzed within and across layers to supports three complementary analyses: cell-type annotation fidelity following SAE reconstruction, the biological relevance and organization of SAE features across layers, and the sensitivity and controllability of C2S model predictions to feature edits. Figure 6 and Appendix B.2 provide a more detailed overview of the pipeline.

collectively termed `PBMC`, provide external evaluation data. Here, "external" refers to the SAE training distribution; exclusion from C2S pretraining has not been verified. Appendix B.1 details the rationale for population selection, data splits, and preprocessing procedures.

2.2 Sparse Autoencoder

For each activation vector $\mathbf{h}^{(\ell)} \in \mathbb{R}^d$, the SAE computes a non-negative sparse latent representation and reconstructs the input as

$$\mathbf{z} = \mathrm{TopK}_k\Big(\mathrm{ReLU}\Big[\mathbf{W}_{\mathrm{enc}}\big(\mathbf{h}^{(\ell)} - \mathbf{b}_{\mathrm{dec}}\big) + \mathbf{b}_{\mathrm{enc}}\Big]\Big), \tag{1}$$

$$\widehat{\mathbf{h}}^{(\ell)} = \mathbf{W}_{\mathrm{dec}}\mathbf{z} + \mathbf{b}_{\mathrm{dec}}, \tag{2}$$

where TopK_k retains the k largest entries and sets the remaining entries to zero, yielding at most k active features per input. Each column of $\mathbf{W}_{\mathrm{dec}}$ defines a feature direction in activation space. We use an expansion factor of 8, yielding 18,432 latent features for the 2B model, with a sparsity budget of $k = 512$. The training objective combines normalized reconstruction error with an auxiliary loss that encourages inactive features to reconstruct the residual $\mathbf{h}^{(\ell)} - \widehat{\mathbf{h}}^{(\ell)}$ (Gao et al., 2025). We constrain each decoder column to have unit ℓ_2 norm, removing the scale ambiguity between latent activations and decoder weights and making latent magnitudes comparable across features.

The implementation details and SAE evaluation metrics are provided in Appendices B.3 and C.1, respectively. Ablation studies guiding architecture and checkpoint selection appear in Appendices D.1 and D.2, with further analyses of the selected checkpoints in Appendix E.

2.3 Feature Interpretation and Activation Editing

Biological feature interpretation. We map genes to SAE features by presenting each gene in the vocabulary in isolation, passing its final-token residual-stream activation through the SAE encoder, and ranking genes by activation magnitude for each feature. Separately, we encode each cell's final-token residual stream at the target layer with the SAE and rank features by their specificity margin: the mean activation within the target cell type minus the largest mean activation among the other cell types. At each analyzed layer, we examine the top 15 features per cell type and manually assess their highest-ranked genes for known cell-type-specific markers, classifying features as cell-type representative or housekeeping-associated. For functional annotation, we submit each selected feature's top 750 genes to g:Profiler for Gene Ontology enrichment, using the full training gene vocabulary as the statistical background (Kolberg et al., 2023). We retain up to ten of the most significant

terms per feature at a Benjamini–Hochberg FDR of ≤ 0.05. Appendix C.2 details the metrics used to quantify feature–cell-type associations.

Reconstruction and steering. We edit only gene-final-token activations during prompt prefill; other token positions and subsequent decoding steps are not directly edited. Reconstruction replaces $\mathbf{h}$ with its SAE reconstruction $\widehat{\mathbf{h}}$. Steering applies a latent edit $\Delta\mathbf{z}$ to either the original or reconstructed activation:

$$\mathbf{h}_{\text{edit}} = \mathbf{h} + \mathbf{W}_{\text{dec}}\Delta\mathbf{z}, \qquad \widehat{\mathbf{h}}_{\text{edit}} = \widehat{\mathbf{h}} + \mathbf{W}_{\text{dec}}\Delta\mathbf{z}.$$

Steer down attenuates selected latent activations by a fraction $\rho \in [0, 1]$, with $\rho = 1$ corresponding to full ablation; *steer up* adds feature-specific positive offsets; and *steer toward* applies both operations simultaneously. We compare interventions at all selected layers, selecting features from the 15 most significant features of each cell type. The total number of edited features varies by intervention. Algorithms 1 and 2 specify the complete procedures.

3 SAE Reconstructions Preserve Annotation Performance

Before evaluating reconstructed embeddings, we verify that the SAEs are well trained by assessing reconstruction fidelity, feature utilization, and decoder diversity. Across all selected layers, the SAEs achieve high reconstruction fidelity on both `purified-PBMC` and the disjoint `PBMC` dataset, with broad dictionary utilization, predominantly selective activations, and minimal feature redundancy. Detailed diagnostics and checkpoint comparisons are provided in Appendix E. These results support using the trained SAEs in the following experiments, where we test whether reconstructed embeddings preserve information relevant to downstream tasks.

SAE reconstruction introduces information loss whose effects can be difficult to characterize while exposing individual features for inspection and intervention (Gao et al., 2025; Engels et al., 2024). We therefore test whether reconstructed embeddings preserve the information needed for the core downstream task, cell-type annotation. Maintaining annotation performance would support their use as a basis for subsequent feature-level analysis and interventions. Following the setup used during SAE training, we pass the target-layer embedding of each gene's final token through the trained SAE, and replace it with its reconstruction before continuing inference (Algorithm 1). This evaluation also tests transfer across prompt contexts: cell-type annotation requires a task-specific prompt prefix, whereas the SAEs were trained on embeddings obtained without this prefix. Thus, the resulting reconstructed representations are task-agnostic rather than explicitly conditioned on the annotation task.

Dataset	Split	n	Base	Layer 10	Layer 15	Layer 20
`Purified-PBMC`	Train	5,395	96.55	**96.74 (+0.19)**	96.57 (+0.02)	96.53 (−0.02)
	Validation	300	95.33	**95.33 (0.00)**	**95.33 (0.00)**	**95.33 (0.00)**
	Test	300	96.33	**96.67 (+0.33)**	96.33 (0.00)	96.33 (0.00)
`PBMC`	Train	3,444	96.40	**97.85 (+1.45)**	96.28 (−0.12)	96.43 (+0.03)
	Validation	430	97.44	**98.84 (+1.40)**	97.44 (0.00)	97.44 (0.00)
	Test	431	96.98	**98.38 (+1.39)**	96.98 (0.00)	96.98 (0.00)

Table 1: **Cell-type annotation accuracy (%) on the train, validation, and test splits of the `purified-PBMC` and `PBMC` datasets.** All splits of the `PBMC` dataset were not involved in SAE training. The 'n' column reports the number of cells in each split. 'Base' reports the performance of the base model using its original representations, whereas the 'Layer ℓ' columns report performance with SAE-reconstructed representations at layer ℓ. Parentheses show percentage-point changes from Base accuracy; bold marks the best SAE result per split (row).

We generate cell-type annotations with greedy decoding. An annotation is scored as correct if it contains the gold cell-type name (B cell, CD14 monocyte, or CD56 NK cell) or an accepted synonym. On both datasets, the SAE reconstructions preserve the annotation performance of the base model: all three layers perform comparably to the original representation, and the occasional slight degradation is too small to indicate a meaningful loss

of predictive information. The results are shown in Table 1. The layer 10 reconstruction consistently has the greatest effect on the model's predictions, whereas layers 15 and 20 are effectively indistinguishable from the baseline. This suggests that by the middle layers, the information relevant to cell-type annotation may already be well established and therefore harder for subsequent reconstruction to alter. The benefit of layer 10 is particularly clear on the PBMC dataset, where it improves on the base prediction across all data splits. This stronger performance on external dataset suggests that the layer-10 reconstruction is robust to the distribution shift and may discard noisy information while retaining features that are relevant for cell-type annotation.

Together, these observations show that SAE reconstruction is at least comparable to the base model for annotation task, with early-middle layers offering a consistent advantage rather than paying a reconstruction cost. We provide more details on the evaluation, error analysis, and model diagnosis for cell-type annotation in Appendix G. We also report supplementary evaluations of a natural-language-based task in Appendix H and explain why we exclude it from the main analysis.

4 Biological Feature Interpretation Across Layers

To understand what biological information SAE features capture and how it varies across layers, we examine two complementary questions: whether selected features distinguish the three cell populations, and whether their associated genes support a biological interpretation. The first assesses feature selectivity within the evaluated populations; the second connects these features to biological concepts through marker and functional associations.

4.1 Cell-Type-Selective Features Emerge with Increasing Layer Depth

Figure 2 summarizes the five highest-ranked features per population at each examined layer (with full heatmaps of the 15 highest-ranked features per population provided in Figure 10), revealing increasingly distinct cell-type selectivity with depth. At layer 10, many selected features remain active across multiple populations, resulting in substantial overlap in their activation patterns. Some features already distinguish individual populations, but separation is uneven, suggesting that early representations contain both selective and broadly shared features.

At layer 15, selected features show more pronounced preferences for their corresponding populations. This separation is particularly apparent among B-cell- and monocyte-selected features, whereas many NK-cell-selected features continue to show appreciable activation in other populations. The intermediate layer thus reflects increasing specialization, with cell-type distinctions becoming clearer at different rates across populations.

At layer 20, the selected features exhibit the clearest separation. Many respond strongly in their associated population and weakly in the others, yielding distinct activation patterns for B cells, monocytes, and NK cells. This sharper selectivity makes features characteristic of each population easier to identify. Nevertheless, some highly ranked features retain substantial activation across populations, indicating that deeper representations continue to combine broadly responsive features with those that are highly selective.

We next examined whether the increasing cell-type selectivity in feature activation was accompanied by clearer organization of feature directions in decoder space. To this end, we jointly embedded the normalized SAE decoder vectors of the 15 highest-ranked features per population at layers 10, 15, and 20 using UMAP with cosine distance (Figure 3). In this shared embedding, features associated with the same cell type tend to form more compact, cohesive groups at deeper layers, while groups associated with different cell types appear more clearly separated. This visual pattern complements the increasingly distinct cell-type selectivity observed in the activation heatmaps.

Together, these findings reveal increasingly distinct cell-type separation in feature activation with depth. Earlier layers show substantial overlap across populations, while deeper layers contain selected features whose activation is more concentrated within individual cell types.

Layer 10

Feature	B	CD14	NK	AUROC
B cell				
F2198	4.2	2.4	1.3	0.645
F5968	8.5	7.5	6.7	0.557
F1206	3.4	2.9	2.0	0.653
F8872	6.8	5.7	6.5	0.565
F16544	9.3	9.0	8.8	0.515
Monocyte				
F16858	4.2	11.1	3.8	0.961
F12530	6.5	10.7	5.6	0.641
F5433	7.6	10.7	6.9	0.621
F12692	9.1	12.1	7.9	0.620
F5414	0.5	3.4	0.4	0.753
NK				
F12445	14.8	9.6	19.2	0.576
F13289	9.4	6.6	12.8	0.701
F5898	1.2	1.6	3.4	0.670
F3423	13.4	9.5	15.2	0.583
F16521	15.4	12.2	16.8	0.583

Layer 15

Feature	B	CD14	NK	AUROC
B cell				
F5475	18.7	7.5	9.0	0.904
F15750	7.8	1.3	0.9	0.894
F1565	7.3	2.4	2.0	0.774
F7819	18.7	9.1	14.7	0.699
F16415	21.0	13.4	17.6	0.630
Monocyte				
F10373	9.8	24.7	6.9	0.821
F10907	16.5	30.5	13.7	0.846
F12837	3.8	17.3	3.7	0.894
F9496	20.5	29.0	19.9	0.651
F10387	7.3	13.1	5.9	0.743
NK				
F4252	8.4	6.4	14.3	0.751
F17222	0.9	1.1	5.5	0.719
F4735	7.3	4.2	11.4	0.696
F5341	18.3	13.1	21.6	0.657
F18065	14.4	9.4	17.2	0.638

Layer 20

Feature	B	CD14	NK	AUROC
B cell				
F17780	31.4	4.7	0.2	0.988
F16995	26.3	0.4	0.9	0.985
F9419	25.2	0.1	0.0	0.960
F18331	21.1	0.7	0.1	0.974
F16617	20.6	0.2	0.5	0.996
Monocyte				
F1146	1.9	36.8	1.9	0.996
F2397	9.1	40.2	12.2	0.924
F15265	0.3	26.2	0.3	0.979
F12201	9.7	32.6	10.7	0.897
F14784	0.1	21.1	0.0	0.987
NK				
F872	0.1	0.2	28.5	0.992
F2262	12.9	10.4	38.8	0.909
F18202	0.0	0.2	18.5	0.974
F803	0.0	0.1	15.9	0.954
F15435	46.1	32.6	59.0	0.669

Figure 2: **Top features relevant to cell types become progressively more specific with layer depth.** The top five cell-type-specific features for each cell type are shown, with the top 15 features displayed in the `purified-PBMC` heatmap (Figure 10). The displayed numbers represent the mean feature activation magnitudes across cells of each cell type (column), with color coding these values. Each AUROC is obtained by using the corresponding activation magnitudes of a feature to classify its associated target cell population against the other two cell populations pooled together.

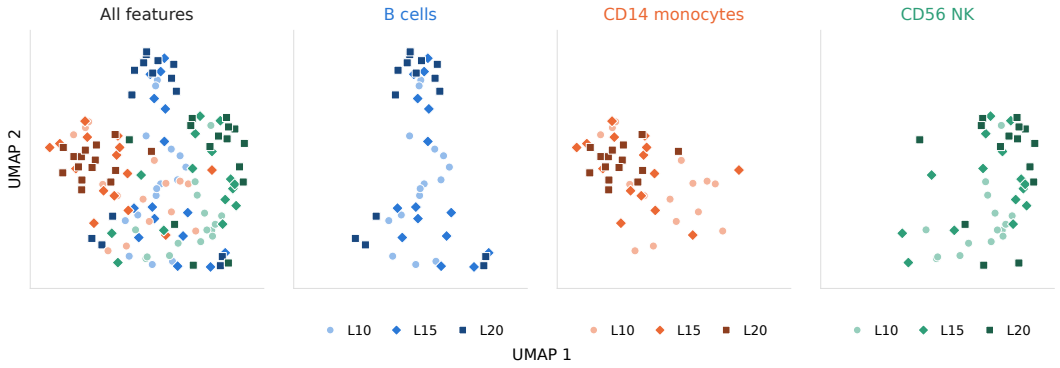


Figure 3: **UMAP visualizations of the top 15 cell-type-relevant SAE features per cell type at layers 10, 15, and 20.** Colors denote cell types; darker shades indicate deeper layers. Panels show all features and features specific to each cell type separately.

4.2 Marker Associations Link Features to Biological Concepts

To examine the biological concepts underlying feature selectivity, we considered associated marker genes alongside GO enrichment. Table 2 illustrates the most representative associations across all selected layers, with full results reported in Tables 14, 15, and 16.

For B and NK cells, the selected examples suggest that later-layer features can capture more lineage-related biological concepts, although this pattern differs between populations. NK-related functions are already evident at layer 15, where the representative enrichment concerns NK-directed recruitment, and at layer 20, where it concerns NK-mediated immunity. The B-cell examples show a less direct progression: at layer 15, B-cell marker associations accompany antigen-presentation machinery shared across populations, whereas the layer-20 example highlights B-cell receptor machinery. This suggests that a recogniz-

Cell	Layer	Feature (rank)	Associated genes	Representative GO term and scope
B	10	**F2198** (1)	*CD19*, *CD22*, *TNFRSF13C*	Lymphocyte activation (shared lymphoid)
	15	**F15750** (2)	*CD19*, *CD22*, *CD79A*	MHC class II protein complex (shared antigen presentation)
	20	**F5600** (12)	*CD79A*, *CD79B*, *EBF1*	IgM B-cell receptor complex (B-cell receptor machinery)
NK	10	**F5898** (3)	*KLRD1*, *CCL5*, *GZMH*	Immune response (shared immune)
	15	**F17222** (2)	*GNLY*, *NKG7*, *CCL5*	Regulation of natural killer cell chemotaxis (NK-directed recruitment)
	20	**F872** (1)	*KLRD1*, *KLRF1*, *GNLY*	Natural killer cell-mediated immunity (NK-related effector function)
CD14	10	**F5414** (5)	*SLC11A1*, *CLEC4E*, *CLEC7A*	Antifungal innate immune response (shared innate/myeloid)
	15	**F14089** (11)	*S100A8*, *S100A9*, *S100A12*	Calprotectin complex (shared myeloid)
	20	**F15694** (13)	*CLEC9A*, *CLEC10A*, *HLA-DPA1*	MHC class II protein complex (shared antigen presentation)

Table 2: **Representative lineage-associated features across layers.** For each cell type at each examined layer (10, 15, or 20), one feature and one of its enriched GO term are selected from the top 15 features by specificity. Feature ranks are shown in parentheses; gene examples are among the top ten associated genes of each feature. B/NK features progress from shared immune to more lineage-related programs, whereas CD14 features retain shared myeloid or antigen-presentation function. Full enrichments are reported in Tables 14, 15, and 16.

able lineage association is present even though the feature's enriched functions have broader biological scope, and are not yet fully cell-type-specific.

Monocytes further illustrate why selectivity within the evaluated populations need not imply biological exclusivity. Across the selected examples, the associated concepts shift from innate immune responses at layer 10 to shared myeloid machinery at layer 15 and antigen presentation at layer 20. The layer-15 example therefore extends, rather than resolves, the broader scope observed at the other layers. Even when later-layer features distinguish monocytes from B and NK cells, their gene associations and enrichments can emphasize programs shared with other myeloid populations. These antigen-presentation associations highlight biological programs shared with dendritic cells, a closely related population within the myeloid lineage. This finding further supports the case-study analysis in Appendix G.2, where misclassifications occur most frequently among monocytes, which are predominantly classified as dendritic cells or underspecified as myeloid cells.

5 Enrichment-Guided Feature Intervention

Having characterized the selectivity and biological associations of SAE features, we next test whether editing these features produces predictable changes in model outputs. These interventions assess whether biologically associated features provide functional control over model predictions. We examine the *steer toward* and *steer up* settings, using additive magnitudes ranging from 20 to 50 and a fixed proportional suppression of 99%. We apply these interventions to both original and SAE-reconstructed activations. Appendix J.1 provides the rationale for this experimental design.

5.1 Layer and Strength Dependence of Feature Steering

We first examine how steering effectiveness varies with intervention depth and additive magnitude under the *steer toward* setting (Figure 4). Steering effects depend strongly on

intervention depth, with layer 10 producing the strongest effects among the three layers examined. This greater effectiveness in early-to-middle layers is consistent with prior findings on contrastive activation and persona steering in language models (Ghandeharioun et al., 2024; Rimsky et al., 2024).

Steering effectiveness also depends on additive magnitude, although the response varies across cell types. At layer 10, larger magnitudes produce stronger steering effects for B cells. For CD14 monocytes, magnitudes above 20 outperform magnitude 20 but yield similar effects to one another, suggesting a plateau within the tested range. In contrast, steering the top-ranked feature for CD56 NK cells at layer 10 does not change the predictions at any tested magnitude. Thus, increasing intervention strength can improve steering, but does not guarantee an effect for every target cell type. Based on these results, we use layer 10 and additive magnitudes ranging from 30 to 50 in subsequent steering experiments.

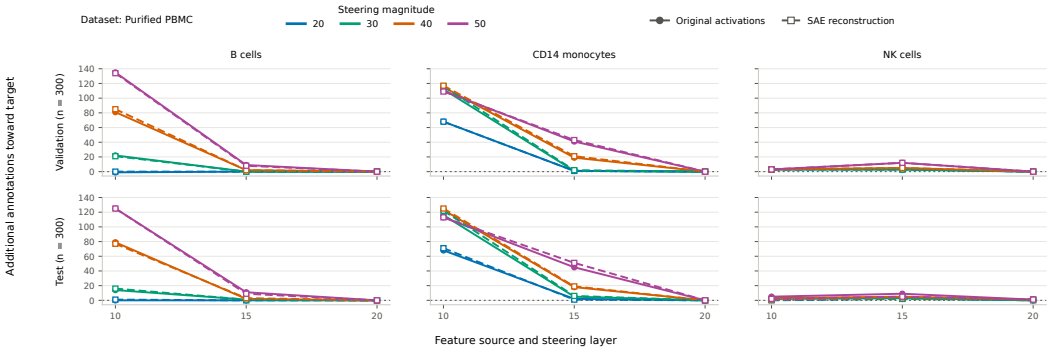


Figure 4: **Layer and steering magnitude dependence of feature steering on cell-type annotation in the `purified-PBMC` dataset.** The y-axis reports the net change in target-class annotations relative to the corresponding unsteered baseline; the gray dashed line marks zero change. Colors indicate additive steering magnitudes under the *steer toward* setting. Solid lines with filled circles and dashed lines with open squares represent the results from using the original activations and SAE reconstructions, respectively. At each layer, steering increases 15 target-associated features and suppresses 30 competing features.

5.2 Cross-Layer SAE Feature Steering Redirects Cell-Type Predictions

The unsuccessful steering of layer-10 *steering toward* CD56 NK cells naturally raised the question: whether this reflected a limitation of NK-directed steering or of the selected feature directions. Although the layer-10 features were selected for their cell-type specificity, selective activation does not necessarily imply effective steering. We therefore investigated cross-layer feature transfer, motivated by the preceding analysis showing more biologically meaningful and cell-type-specific features in later layers. Because cross-layer feature directions do not directly correspond to feature activations at the intervention layer, we restrict cross-layer steering experiments to *steer up* mode. As shown in Figure 5, keeping the intervention fixed at layer 10 while using later-layer features substantially strengthened NK-directed steering, with layer-15 features producing the strongest effect. Layer-15 features produced the strongest target-directed steering toward both CD56 NK cells and CD14 monocytes across both datasets. B-cell steering was the exception: the most effective source layer varied with the dataset and steering magnitude. We hypothesize that these differences reflect how clearly features at each layer correspond to cell-type-specific biological concepts, and we further examine this hypothesis in Appendix J.2 through annotation transitions and GO-enrichment patterns.

These findings show that biologically associated features can provide effective directions for steering predictions toward the corresponding cell types, supporting their interpretation as representations of cell-related information that contributes to model behavior. Their effectiveness nevertheless depends on both the feature source and intervention layers: later-

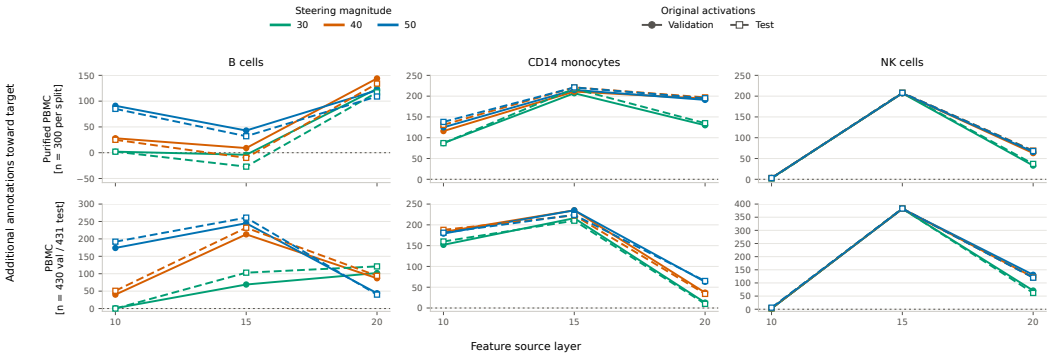


Figure 5: **Cross-layer feature steering of cell-type annotations.** SAE features from layers 10, 15, and 20 are used to steer activations at layer 10. Colors indicate steering magnitude; solid and dashed lines denote validation and test results. The y-axis shows the net change in target-class annotations relative to the unsteered model, with panel-specific ranges. Later-layer features can strengthen steering, depending on the target cell type and dataset.

layer features can successfully steer predictions at an earlier layer, while activation selectivity alone does not establish steering effectiveness.

5.3 Reconstruction Preserves Observed Steering Patterns

Finally, we examine whether the observed steering responses persist when the original hidden states are replaced by their SAE reconstructions. Across the evaluated conditions, reconstruction preserves the main patterns of intervention sensitivity, with differences between the two steering bases remaining small relative to those across different steering protocols.

These results suggest that SAE reconstruction retains sufficient information to reproduce the observed annotation-level responses to the tested interventions. In particular, the strong response at layer 10 and weaker responses at later layers persist under reconstruction. This agreement further supports the view that SAE reconstruction preserves cell-related information in the model and that reconstructed embeddings retain functional alignment with the original hidden states under the tested interventions.

6 Conclusion and Future Work

We present an SAE-based framework that links representations in cross-modal scFMs to biological knowledge and evaluates their influence via cell-type annotations. SAE-reconstructed activations preserve annotation performance comparable to the original activations, supporting the use of SAEs as translators from opaque embeddings to more interpretable representations. Targeted steering experiments show that learned features influence model predictions, with effectiveness varying across feature source layers and target cell types. Together with GO-enrichment terms, these results suggest that stronger target-directed steering may be associated with clearer links to cell-type-specific biological concepts. Feature-based error analysis further yields testable hypotheses about the biological distinctions underlying steering-induced annotation shifts. By combining reconstruction-based performance checks, biological interpretation, and feature steering, our framework provides a systematic approach to investigating how learned biological information contributes to model behavior.

Future work can extend this framework to additional cross-modal foundation models trained under different settings and evaluate its generalizability across larger and more diverse biological datasets. These efforts could help reveal how biological concepts are encoded, organized, and utilized to support a wide spectrum of downstream tasks, while further quantifying uncertainty to support reliable predictions and inform the development of the next generation of more powerful scFMs.

A Related Work

A.1 Language Foundation Models for Single-Cell Biology

Single-cell analysis has become central to modern biology because it resolves distinct cells, the resolution where biological processes actually occur. In parallel, deep learning has made it possible to learn patterns from large, complex biological datasets and build predictive models of living systems. The rapid expansion of single-cell atlases across tissues, diseases, species, modalities, and perturbations provides the data foundation for single-cell foundation models: large-scale models that learn general representations of cells and predict cell states, gene regulation, differentiation, and perturbation responses.

Single-cell foundation models fall into two broad categories. The first, expression-native models, are trained directly on large single-cell gene-expression corpora. Models such as Geneformer (Theodoris et al., 2023), scGPT (Cui et al., 2024), scFoundation (Hao et al., 2024) learn cellular representations from transcriptomic profiles, typically treating genes or expression values as inputs and optimizing masked-expression prediction, reconstruction, metric-learning, or perturbation objectives. They have proven broadly useful for cell-type annotation, batch integration, and perturbation prediction, though their biological knowledge remains largely implicit in the training data and objective. The second category links cells with natural-language biological knowledge. Models such as GenePT (Chen & Zou, 2024), LangCell (Zhao et al., 2024), InstructCell (Fang et al., 2025), scMMGPT (Shi et al., 2025), OKR-CELL (Wang et al., 2026), and Cell2Sentence (Rizvi et al., 2026) integrate transcriptomic profiles with textual information through pretrained language models, cross-modal alignment, or language-based representations of cells. This integration allows the models to incorporate semantic information about gene functions, biological pathways, cell identities, and disease contexts beyond expression values alone. Consequently, they can support tasks such as open-vocabulary cell-type annotation, cell–text retrieval, and natural-language analysis while drawing on the broad biomedical knowledge encoded in language models.

A.2 Biological Interpretability and Concept Discovery

Interpretability has been a persistent but under-served need throughout the development of biological foundation models. As these models are increasingly used not merely for prediction but as instruments for scientific discovery, understanding *what* they encode internally becomes as important as *how well* they perform. Biology is also an unusually favorable setting for mechanistic interpretation: unlike natural language, many of its concepts are externally grounded in curated ontologies and annotations (e.g., binding sites, functional domains, gene programs), offering objective targets against which learned representations can be validated. The mechanistic interpretability toolkit developed for language models therefore transfers naturally to this domain: probing, which tests whether a target concept is linearly decodable from a model's internal representations; SAEs (Cunningham et al., 2023), which decompose dense activations into sparse, human-interpretable features and thereby disentangle concepts held in superposition; and transcoders, which approximate a layer's computation with sparse features to support circuit discovery and feature interpretability. Protein language models were the first to adopt this toolkit, where SAEs have recovered features that align with known structural and functional annotations (Simon & Zou, 2025; Adams et al., 2025). Single-cell analysis, by contrast, has seen comparatively few such attempts, despite its popularity and a rapidly growing collection of foundation models. A handful of recent studies apply SAEs to *expression-native* scFMs, training dictionaries on the hidden representations of models to surface biological and technical signals and, in some cases, to steer model behavior (Schuster, 2025; Claye et al., 2025; Pedrocchi et al., 2025). The *natural-language* branch of single-cell modeling, however, remains largely unexplored. Motivated by this gap, we study the interpretability of C2S directly, asking whether its internal representations admit a coherent biological interpretation and whether its learned features encode identifiable biological concepts.

B Implementation Details

B.1 Cell-Type Selection and Data Preparation

We focus feature-level analysis on three immune cell populations—B cells, CD14 monocytes, and CD56 natural killer cells—which jointly span the lymphoid and myeloid branches of hematopoiesis and the adaptive and innate arms of immunity, so that a concept shared across all three is unlikely to be a lineage artifact. All three are among the FACS-sortable populations used to establish reference PBMC transcriptomes and are recovered as major, well-separated clusters at expected proportions from expression alone (Zheng et al., 2017), giving both high-confidence labels and enough cells per type for stable activation-frequency and enrichment statistics. We deliberately avoid finer distinctions such as naive CD4 versus CD8 T cells, whose transcriptomes overlap to the point that surface-protein or chromatin measurements are needed to resolve them reliably (Hao et al., 2021); attributing a latent to one of those subsets would be confounded by shared programs in a way we cannot control.

For each selected immune cell type, we randomly sampled 2,000 cells from the `purified-PBMC` dataset and split them into training, validation, and test sets at a ratio of 90%/5%/5%. Only the `purified-PBMC` data were used for model training and validation. All cells from the `PBMC-4k` and `PBMC-8k` datasets were reserved for external testing; for some downstream tasks, however, they were partitioned into training, validation, and test sets using an 80%/10%/10% split. We then apply standard quality control: cells are retained if they express between 200 and 2,000 genes and carry less than 20% mitochondrial counts, and genes are retained if detected in at least three cells. Counts are normalized per cell to a common total and then log-transformed, following the C2S convention; gene symbols are uppercased and deduplicated. Crucially, the gene vocabulary is constructed from the training split alone, and any validation or test cell containing an out-of-vocabulary gene is discarded before extraction; every downstream split is verified against this vocabulary, so no held-out gene identity is ever observed at training time.

B.2 Overview of the C2S–SAE Pipeline

Figure 6 details the workflow summarized in Figure 1 from single-cell expression profiles to interpretable SAE features. After preprocessing, genes are ranked by expression to form an ordered cell sentence, which is tokenized and passed through the frozen C2S model. At each examined layer, we use the final-token activation of each gene as its representation and train a separate TopK SAE on the collected activations. The SAE decomposes these representations into sparse combinations of learned feature directions. We characterize the resulting features through their activation patterns across cell types and Gene Ontology enrichment, and evaluate their influence on model predictions through reconstruction and steering experiments.

B.3 SAE Architecture and Training

Architecture and initialization. Each activation $\mathbf{h}^{(\ell)}$ is first centered by subtracting the decoder bias $\mathbf{b}_{\text{dec}}$. The encoder maps the centered activation to latent pre-activations, after which a TopK gate retains the k largest values to form the sparse feature activations $\mathbf{z}$. The decoder then reconstructs the original activation and adds $\mathbf{b}_{\text{dec}}$ back as an offset. Formally, the process can be described as:

$$\widetilde{\mathbf{h}}^{(\ell)} = \mathbf{h}^{(\ell)} - \mathbf{b}_{\text{dec}}, \qquad \mathbf{z} = \text{TopK}_k\left(\text{ReLU}\left(\mathbf{W}_{\text{enc}}\widetilde{\mathbf{h}}^{(\ell)} + \mathbf{b}_{\text{enc}}\right)\right), \qquad \widehat{\mathbf{h}}^{(\ell)} = \mathbf{W}_{\text{dec}}\mathbf{z} + \mathbf{b}_{\text{dec}}.$$

The columns of the decoder weight matrix $\mathbf{W}_{\text{dec}}$ constitute the learned feature dictionary, with each column representing one latent feature direction. The decoder bias is initialized to the mean activation of the first batch. Before each optimization step, each decoder column is normalized to unit ℓ_2 norm, and the component of its gradient parallel to that column is removed to account for the interaction between Adam and decoder normalization (Bricken et al., 2023; Gao et al., 2025).

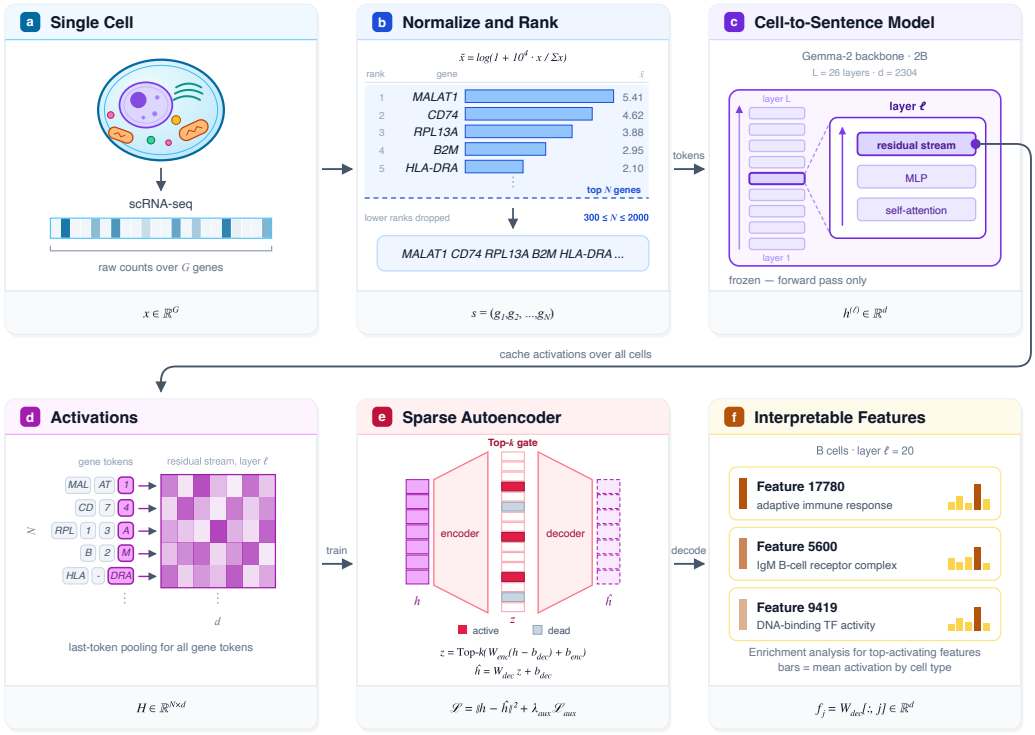


Figure 6: **C2S–SAE pipeline for interpreting gene-level representations.** **(a)** Each cell is represented by a vector of raw scRNA-seq counts over G genes. **(b)** Counts are library-size normalized and log-transformed, after which genes are ranked by expression and the top N genes are retained to form an ordered "cell sentence." **(c)** The cell sentence is tokenized and passed through a frozen C2S model with a Gemma-2 backbone. Residual-stream activations are extracted from a selected transformer layer ℓ and cached across cells. **(d)** Because a gene name may span multiple tokens, the activation of its final token is used as its gene-level representation. Stacking the representations of all retained genes produces an activation matrix $\mathbf{H} \in \mathbb{R}^{N \times d}$. **(e)** A TopK sparse autoencoder is trained on these activations. Each activation $\mathbf{h}$ is centered using the decoder bias, encoded into a sparse latent representation $\mathbf{z}$, and reconstructed as $\widehat{\mathbf{h}}$. The training objective combines reconstruction error with an auxiliary loss that provides a learning signal to dead features. **(f)** Each column of the decoder weight matrix defines a learned feature direction. Representative features with high activation in B cells, obtained from an SAE trained on layer-20 activations of C2S Gemma-2-2B, are annotated using Gene Ontology enrichment; the bars show their mean activation across cell types.

Training objective. The SAE is trained using a normalized reconstruction loss together with an auxiliary loss:

$$\mathcal{L} = \mathcal{L}_{\text{rec}} + \lambda_{\text{aux}} \mathcal{L}_{\text{aux}}.$$

The reconstruction loss measures the mean squared error between $\mathbf{h}^{(\ell)}$ and $\hat{\mathbf{h}}^{(\ell)}$, normalized by the variance of the input activations. The auxiliary loss uses features classified as "dead" according to an exponential moving average of feature activity to reconstruct the residual $\mathbf{h} - \hat{\mathbf{h}}$ (Morales-Brotons et al., 2024). Because sparsity is imposed directly by the TopK gate, no additional sparsity penalty is introduced.

Hyperparameters and computational setup. The key hyperparameters used for SAE training are summarized in Table 3. The encoder-decoder weights are untied. No learning-

rate schedule, warmup, gradient clipping, or weight decay was applied. All models were trained on NVIDIA GB10 GPU with CUDA 13.0.

Category	Hyperparameter	Value
Architecture	SAE type	Top-K
	Expansion (α)	8
	Latent width	$\alpha \cdot d_{\text{model}}$
Sparsity	Active latents (k)	512
	Auxiliary width (k_{aux})	$4 \times k$
	Auxiliary-loss weight	1/32
	Firing-rate EMA decay	0.99
Optimization	Optimizer	Adam
	Learning rate	1×10^{-4}
	Adam ϵ	6.25×10^{-10}
	Weight decay	0
	Batch size	4096
	Precision	bfloat16 mixed precision
Training	Maximum epochs	50
	Early-stopping patience	5 epochs
	Minimum improvement	1×10^{-4}
	Monitored metric	Validation reconstruction FVU

Table 3: **Hyperparameters for SAE training.**

C Metrics

C.1 SAE Metrics

We evaluate SAE on a corpus of C cells. Each cell c is represented by a cell sentence

$$s_c = (g_{c,1}, \ldots, g_{c,N_c}),$$

and contributes one activation per gene, taken at the gene's final token. The corpus therefore yields $n = \sum_{c=1}^{C} N_c$ gene-level activations $\mathbf{h}_i \in \mathbb{R}^d$ at layer ℓ. When the cell of origin is relevant, we write these activations as $\mathbf{h}_{c,t}$. We index activations by i and, since the layer is fixed throughout this section, abbreviate $\mathbf{h}_i \equiv \mathbf{h}_i^{(\ell)}$.

Applying the SAE encoder to $\mathbf{h}_i$ produces the feature activations $\mathbf{z}_i \in \mathbb{R}^D$, the reconstruction $\widehat{\mathbf{h}}_i$, and the residual

$$\mathbf{r}_i = \mathbf{h}_i - \widehat{\mathbf{h}}_i.$$

We index features by $j \in \{1, \ldots, D\}$ and write $z_{ij} = [\mathbf{z}_i]_j$ for the activation of feature j on $\mathbf{h}_i$, or $z_{c,t,j}$ when the cell–gene structure is relevant. Feature j is *active* on activation i if $z_{ij} > 0$. The firing set and activation frequency of $\mathbf{h}_i$ are

$$\mathcal{S}_j = \{i : z_{ij} > 0\}, \qquad f_j = \frac{|\mathcal{S}_j|}{n}.$$

The Top-k gate ensures that exactly k features are active for every i, i.e.,

$$\big|\{j : z_{ij} > 0\}\big| = k.$$

Reconstruction error. We measure reconstruction error using root mean squared error (RMSE),

$$\text{RMSE} = \sqrt{\frac{1}{nd} \sum_{i=1}^{n} \|\mathbf{h}_i - \hat{\mathbf{h}}_i\|_2^2}, \tag{3}$$

where n is the number of samples, d is the activation dimension, and $\mathbf{r}_i = \mathbf{h}_i - \hat{\mathbf{h}}_i$ is the reconstruction residual. Because RMSE depends on the activation scale and is therefore not

directly comparable across layers, we additionally report normalized mean squared error (NMSE),

$$\text{NMSE} = \frac{\sum_i \|\mathbf{r}_i\|_2^2}{\sum_i \|\mathbf{h}_i\|_2^2}. \tag{4}$$

For both metrics, a smaller value corresponds to a more accurate reconstruction.

Variance explained. While RMSE and NMSE quantify reconstruction error, the fraction of variance unexplained (FVU) measures the residual variation relative to the variation in the original activations:

$$\text{FVU} = \frac{\sum_i \|\mathbf{r}_i - \bar{\mathbf{r}}\|_2^2}{\sum_i \|\mathbf{h}_i - \bar{\mathbf{h}}\|_2^2} = \frac{\sum_{j=1}^{d} \text{Var}(r_{\cdot j})}{\sum_{j=1}^{d} \text{Var}(h_{\cdot j})}. \tag{5}$$

We report the corresponding fraction of variance explained,

$$\text{FVE} = 1 - \text{FVU}. \tag{6}$$

A smaller FVU, and therefore a larger FVE, indicates that less of the original variation remains unexplained by the reconstruction.

Because aggregate FVE gives greater weight to high-variance dimensions, we also compute the mean FVE across individual dimensions:

$$\text{FVE}_{\text{dim}} = \frac{1}{|\mathcal{J}|} \sum_{j \in \mathcal{J}} \left(1 - \frac{\text{Var}(r_{\cdot j})}{\text{Var}(h_{\cdot j})}\right), \tag{7}$$

where $\mathcal{J}$ excludes dimensions with zero variance.

Activation norms also vary substantially across gene tokens, allowing high-norm samples to have a disproportionate influence on the aggregate metrics. We account for this variation by normalizing each activation and its residual by the activation norm

$$\widetilde{\mathbf{h}}_i = \frac{\mathbf{h}_i}{\|\mathbf{h}_i\|_2}, \qquad \widetilde{\mathbf{r}}_i = \frac{\mathbf{r}_i}{\|\mathbf{h}_i\|_2}, \tag{8}$$

and recomputing FVE on the normalized vectors to obtain FVE_{norm}. The difference between FVE and FVE_{norm} indicates how strongly the aggregate result is influenced by high-norm samples.

Feature utilization. The activation frequency of feature j is the fraction of gene activations on which it fires:

$$f_j = \frac{|\mathcal{S}_j|}{n} = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}[z_{ij} > 0]. \tag{9}$$

For features that fire at least once, we define the mean activation when active as

$$\mu_j^{\text{active}} = \frac{1}{|\mathcal{S}_j|} \sum_{i \in \mathcal{S}_j} z_{ij} = \frac{\sum_{i=1}^{n} z_{ij} \mathbb{1}[z_{ij} > 0]}{\sum_{i=1}^{n} \mathbb{1}[z_{ij} > 0]}. \tag{10}$$

This quantity is related to the mean activation magnitude defined above by $\mu_j = f_j \mu_j^{\text{active}}$.

Summing the activation frequencies across features gives the mean number of active features per gene activation:

$$\sum_{j=1}^{D} f_j = \frac{1}{n} \sum_{i=1}^{n} \|\mathbf{z}_i\|_0 = \overline{L}_0. \tag{11}$$

When the TopK operator produces exactly k nonzero activations for every $\mathbf{z}_i$, we have $\overline{L}_0 = k$.

We classify feature j as *dead* if $f_j < 10^{-4}$, *common* if $f_j \geq 0.05$, and *universal* if $f_j \geq 0.5$. These categories are nested, so every universal feature is also common. To quantify imbalance in feature utilization, we compute the Gini coefficient of the activation frequencies:

$$G = \frac{\sum_{j=1}^{D} \sum_{j'=1}^{D} |f_j - f_{j'}|}{2D \sum_{j=1}^{D} f_j} = \frac{\sum_{j=1}^{D} \sum_{j'=1}^{D} |f_j - f_{j'}|}{2D\overline{L}_0}. \tag{12}$$

A value of $G = 0$ indicates uniform utilization across the dictionary, whereas a value closer to 1 indicates that activations are concentrated among a relatively small subset of features.

Decoder geometry. Let $\mathbf{w}_j$ be decoder direction j. Each feature's nearest-neighbor score is the absolute cosine similarity between feature j and its nearest decoder neighbor

$$s_j = \max_{\ell \neq j} \left| \frac{\mathbf{w}_j^\top \mathbf{w}_\ell}{\|\mathbf{w}_j\|_2 \|\mathbf{w}_\ell\|_2} \right|. \tag{13}$$

This score captures local crowding in the decoder dictionary: low values indicate well-separated feature directions, whereas high values indicate that many features have nearly parallel or anti-parallel alternatives and may therefore be geometrically redundant. We report the median of s_j to characterize the directional similarity among decoder features. Note that since s_j uses absolute cosine, it measures directional redundancy rather than semantic equivalence.

PCA comparison. We evaluate whether SAE features merely recover the dominant dense directions identified by PCA. The first evaluation compares reconstruction quality under a matched direction budget. At budget q, PCA reconstructs each activation using a single global rank-q subspace, whereas the SAE selects the q strongest decoder atoms separately for each sample. Both reconstructions are evaluated using the same centered FVE, with

$$\Delta_{\mathrm{PCA}}(q) = \mathrm{FVE}_{\mathrm{SAE}}(q) - \mathrm{FVE}_{\mathrm{PCA}}(q). \tag{14}$$

The primary operating point is $q = 512$, matching the k used during SAE training; comparisons at $q < 512$ are supplementary because truncating a trained Top-k SAE is off-distribution. This comparison matches the number of directions used to reconstruct each sample, yet SAE remains more flexible because it selects those directions from a larger dictionary separately for each sample. We additionally evaluate decoder–PCA alignment to determine whether the SAE merely reproduces the dominant directions identified by PCA or learns distinct directions. For each decoder direction, we compute its maximum absolute cosine similarity with the leading PCs and compare this overlap against a random-unit-vector null; systematically high overlap indicates recovery of PCA-like axes, whereas chance-level overlap suggests directions not reducible to any single leading PC.

C.2 Feature Metrics

We evaluate features through their activation strength, cell-type specificity, and association with cell type annotation errors. Let $\mathbf{h}_c \in \mathbb{R}^d$ be the final representation of cell c. Applying the SAE encoder gives $\mathbf{z}_c \in \mathbb{R}^D$, where z_{cj} denotes the activation of feature j for that cell. Let y_c and $\hat{y}_c$ denote the gold and predicted cell types, respectively. We write $\mathcal{I}_a = \{c : y_c = a\}$ for the set of cells of type a, with $n_a = |\mathcal{I}_a|$.

Mean activation and activation frequency. With unit-norm decoder directions $\mathbf{w}_j$, the reconstruction relative to the decoder bias is

$$\widehat{\mathbf{h}}_i - \mathbf{b}_{\mathrm{dec}} = \sum_{j=1}^{D} z_{ij} \mathbf{w}_j. \tag{15}$$

Because $z_{ij} \geq 0$ and $\|\mathbf{w}_j\|_2 = 1$, the magnitude of feature j's contribution is z_{ij}. We therefore measure its average contribution across gene-level activations using

$$\mu_j = \frac{1}{n} \sum_{i=1}^{n} z_{ij}. \tag{16}$$

As defined above, f_j is the fraction of gene-level activations on which feature j fires, and μ_j^{active} is its mean activation conditional on firing. These quantities satisfy $\mu_j = f_j \mu_j^{\mathrm{active}}$, separating how often a feature fires from how strongly it contributes when active.

Specificity margin. To measure whether a feature is selective for a particular cell type, we first compute its mean cell-level activation within that type:

$$\mu_j^{(a)} = \frac{1}{n_a} \sum_{c \in \mathcal{I}_a} z_{cj}. \tag{17}$$

We define the specificity margin as the difference between this mean and the largest mean among the other cell types:

$$m_{j,a} = \mu_j^{(a)} - \max_{a' \neq a} \mu_j^{(a')}. \tag{18}$$

A positive margin indicates that feature j has a higher mean activation in type a than in every other type. This comparison is stricter than comparing against the pooled mean of all other cells: a feature shared equally by two related cell types has a zero margin even if it separates both from the remaining types. For each cell type, we rank features by decreasing $m_{j,a}$, break ties by decreasing $\mu_j^{(a)}$, and retain the top features with positive margins.

Cell-type annotation AUROC. We measure whether feature activation distinguishes incorrectly annotated cells from correctly annotated cells of the same gold type. For each type a, we define the correct and failure groups as

$$\mathcal{C}_a = \{c \in \mathcal{I}_a : \hat{y}_c = y_c\}, \qquad \mathcal{F}_a = \{c \in \mathcal{I}_a : \hat{y}_c \neq y_c\}, \tag{19}$$

and when analyzing specific error categories, we restrict $\mathcal{F}_a$ to failures in those categories.

We treat failure as the positive class and use z_{cj} as the score, and compute AUROC using the Mann–Whitney U statistic, assigning average ranks to tied scores. For independently and uniformly sampled cells $c_F \in \mathcal{F}_a$ and $c_C \in \mathcal{C}_a$, we define

$$\mathrm{AUROC}_{j,a} = \Pr(z_{c_F j} > z_{c_C j}) + \frac{1}{2} \Pr(z_{c_F j} = z_{c_C j}). \tag{20}$$

AUROC measures how often a failed annotation receives a higher feature score than a correct annotation, with ties receiving half credit. Values above 0.5 indicate higher activation in failures, whereas values below 0.5 indicate higher activation in correct predictions. A value of 0.5 indicates no overall ranking preference between the two groups.

Standardized effect size (Cohen's d). Alongside AUROC, which measures rank separation, we report Cohen's d to quantify the difference in mean feature activation relative to within-group variability. For each gold cell type a, we use the same failure and correct groups, $\mathcal{F}_a$ and $\mathcal{C}_a$, as in the AUROC analysis. Let $n_F = |\mathcal{F}_a|$ and $n_C = |\mathcal{C}_a|$, with group means $\mu_{j,F}$ and $\mu_{j,C}$ of z_{cj}, and unbiased sample variances $s_{j,F}^2$ and $s_{j,C}^2$. We omit their dependence on a for readability. The pooled standard deviation and effect size are

$$s_{j,a} = \sqrt{\frac{(n_F - 1)s_{j,F}^2 + (n_C - 1)s_{j,C}^2}{n_F + n_C - 2}}, \qquad d_{j,a} = \frac{\mu_{j,F} - \mu_{j,C}}{s_{j,a}}. \tag{21}$$

Positive values indicate higher mean activation in failures, whereas negative values indicate higher mean activation in correct predictions. We rank features by $|d_{j,a}|$ when direction is not of interest. When $s_{j,a} = 0$, we assign $d_{j,a} = 0$ by convention, although the effect size is undefined and this value does not imply equal group means.

D Ablation Studies for SAEs

D.1 Architecture and Sparsity

We perform a hyperparameter sweep to choose a configuration that reconstructs the activations well while keeping the learned features interpretable. Allowing more latents to be active improves reconstruction, but beyond a point the features become too fine-grained to interpret; the sweep locates the setting where reconstruction is adequate before that happens. We fix the dictionary width to a fixed expansion factor $\alpha \in \{4, 8\}$ over the residual

stream dimension d, and vary the Top-k sparsity level over $k \in \{128, 256, 512\}$. We conduct the hyperparameter sweep using layer-15 activations and evaluate all configurations at epoch 10, with the results reported in Table 4.

The $\{\alpha = 8,\ k = 512\}$ configuration provides the best overall SAE because it achieves the strongest reconstruction performance, with the highest FVE and dimension-wise FVE, alongside the lowest MSE and relative L_2 error. It also exhibits the least redundant decoder geometry, having the lowest nearest-neighbor cosine similarity and proportion of highly similar feature pairs, suggesting that its learned features are more distinct rather than duplicated. From a biological perspective, $k = 512$ is preferable to $k = 256$ because the larger active-feature budget provides sufficient capacity to separate biological programs that may co-occur or have correlated expression patterns; with $k = 256$, such programs may instead be compressed into mixed or incompletely resolved features. At the same time, $k = 512$ remains sparse enough to avoid distributing biological signals across an excessively large number of features, which would fragment coherent programs and make the representation difficult to interpret. Thus, the hyperparameter combination $\{\alpha = 8,\ k = 512\}$ offers a favorable balance between reconstruction fidelity, feature distinctness, biological resolution, and interpretability.

Hyperparams		Reconstruction				Feature usage			Decoder geometry	
α	k	FVE (%)	Dim. FVE (%)	MSE	Rel. L_2 (%)	Universal (%)	Common (%)	Gini	NN $\lvert\cos\rvert$	Similar NN (%)
4	128	91.33	89.20	1.1667	20.26	0.02	5.78	0.6987	0.1717	0.63
4	256	92.82	91.01	0.9665	18.44	0.17	10.51	0.5817	0.1445	0.63
4	512	94.56	93.18	0.7319	16.05	1.76	22.90	**0.5422**	0.1241	0.41
8	128	92.20	90.29	1.0493	19.22	0.00	2.26	0.7302	0.1689	0.27
8	256	93.52	91.89	0.8722	17.52	0.14	3.50	0.6031	0.1413	0.35
8	**512**	**95.05**	**93.77**	**0.6670**	**15.32**	0.93	5.77	0.5547	**0.1192**	**0.14**

Table 4: **Layer-15 TopK SAE hyperparameter comparison at epoch 10.** Reconstruction metrics are evaluated on the held-out test split, feature statistics use the combined splits. α denotes the SAE expansion factor, with the latent dimension $d_{\text{latent}} = \alpha \cdot d_{\text{model}}$. Universal and common features have activation rates of at least 50% and 5%, respectively. Dim. FVE is the mean per-dimension variance explained. NN $\lvert\cos\rvert$ is the mean nearest-neighbor absolute decoder cosine similarity. Similar NN is the fraction of features whose nearest decoder neighbor has $\lvert\cos\rvert \geq 0.8$.

D.2 Epoch Selection

Large-scale SAEs for language models are commonly trained on billions of activation tokens, with some prominent implementations using only a single epoch over the activation dataset (Bricken et al., 2023; Gao et al., 2025; Lieberum et al., 2024). Because the single-cell dataset is more scarce and contains recurring gene-expression patterns, we instead train for multiple epochs and examine intermediate checkpoints to characterize the tradeoff between reconstruction fidelity and dictionary health.

Table 5 shows a consistent trajectory across all three residual-stream layers. Reconstruction improves rapidly during the early stages of training and then begins to plateau. By epoch 10, the SAEs already reconstruct the residual stream well and outperform the matched PCA baseline at every layer. Further training continues to improve reconstruction, but the gains become increasingly modest and accompanied by a less balanced dictionary. Feature usage becomes more concentrated, universal and duplicate features become more prevalent, and dead features eventually emerge in the deepest layer. This pattern is especially pronounced in the middle and deeper layers, where substantial concentration is already visible relatively early in training.

We therefore select epoch 10 for feature-level analysis. It is the earliest evaluated checkpoint that achieves strong reconstruction and consistently exceeds the PCA baseline across all layers, while avoiding some of the additional concentration and redundancy that develop during later training. This choice should be understood as a practical compromise rather than evidence that epoch-10 features are intrinsically the most interpretable. Later checkpoints may

still be preferable when reconstruction fidelity is more important than maintaining a broad and balanced feature dictionary.

C2S Gemma-2B SAEs ($D = 18{,}432$, $k = 512$)

Panel A: reconstruction and PCA comparison

Checkpoint	FVE ↑	FVE_{norm} ↑	CosSim ↑	NMSE ↓	$\Delta_{\text{PCA}}(512)$ ↑
Layer 10					
Epoch 1	0.9202	0.9109	0.9754	0.0449	0.0231
Epoch 10	0.9759	0.9733	0.9927	0.0136	0.0788
Epoch 20	0.9797	0.9775	0.9938	0.0114	0.0826
Epoch 30	0.9812	0.9791	0.9943	0.0106	0.0840
Epoch 40	0.9819	0.9800	0.9945	0.0102	0.0848
Last (50)	**0.9824**	**0.9805**	**0.9946**	**0.0099**	**0.0853**
Layer 15					
Epoch 1	0.8813	0.8775	0.9711	0.0562	−0.0076
Epoch 10	0.9505	0.9490	0.9880	0.0235	0.0616
Epoch 20	0.9578	0.9565	0.9898	0.0200	0.0688
Epoch 30	0.9604	0.9592	0.9904	0.0188	0.0715
Epoch 40	0.9618	0.9606	0.9908	0.0181	0.0729
Last (50)	**0.9627**	**0.9616**	**0.9910**	**0.0177**	**0.0738**
Layer 20					
Epoch 1	0.9247	0.9102	0.9855	0.0236	−0.0331
Epoch 10	0.9762	0.9710	0.9953	0.0074	0.0185
Epoch 20	0.9805	0.9762	0.9962	0.0061	0.0228
Epoch 30	0.9820	0.9780	0.9965	0.0056	0.0243
Epoch 40	0.9828	0.9790	0.9966	0.0054	0.0251
Last (50)	**0.9833**	**0.9796**	**0.9967**	**0.0052**	**0.0256**

Panel B: feature-dictionary health

Checkpoint	Dead % ↓	Common %	Universal % ↓	Gini ↓	Duplicate % ↓
Layer 10					
Epoch 1	**0.00**	5.93	**0.00**	**0.2934**	**0.00**
Epoch 10	**0.00**	5.47	0.21	0.3494	0.01
Epoch 20	**0.00**	6.13	0.39	0.4103	0.02
Epoch 30	**0.00**	6.32	0.46	0.4458	0.03
Epoch 40	**0.00**	6.53	0.53	0.4692	0.05
Last (50)	**0.00**	6.56	0.57	0.4869	0.07
Layer 15					
Epoch 1	**0.00**	6.22	**0.00**	**0.3561**	**0.00**
Epoch 10	**0.00**	5.77	0.93	0.5566	0.08
Epoch 20	**0.00**	7.87	1.04	0.6566	0.27
Epoch 30	**0.00**	8.71	1.03	0.6964	0.41
Epoch 40	**0.00**	9.22	1.04	0.7193	0.58
Last (50)	**0.00**	9.55	1.03	0.7341	0.78
Layer 20					
Epoch 1	**0.00**	5.92	**0.04**	**0.4371**	**0.00**
Epoch 10	**0.00**	8.36	1.25	0.7527	0.14
Epoch 20	0.13	8.82	1.37	0.8165	0.34
Epoch 30	0.38	8.90	1.41	0.8320	0.42
Epoch 40	0.80	8.90	1.43	0.8378	0.54
Last (50)	1.01	8.87	1.41	0.8404	0.76

Table 5: **SAE checkpoint sweep across the three C2S Gemma-2-2B residual-stream layers.** Additional training improves reconstruction at every layer but progressively concentrates feature use. Panel A reports held-out reconstruction and the matched-budget PCA comparison; Panel B reports feature-dictionary health. All values are evaluated on test split of `purified-PBMC` dataset. Bold marks the best value within each layer block.

	Reconstruction			Feature-dictionary health				
Layer	FVE ↑	FVE_{norm} ↑	NMSE ↓	Dead % ↓	Common %	Univ. % ↓	Gini ↓	Dup. % ↓
`purified-PBMC`								
10	0.9763	**0.9737**	0.0134	**0.00**	5.48	**0.21**	**0.3502**	**0.01**
15	0.9518	0.9504	0.0228	**0.00**	5.77	0.93	0.5547	0.08
20	**0.9780**	0.9730	**0.0068**	**0.00**	8.40	1.25	0.7514	0.14
`PBMC`								
10	0.9434	**0.9430**	0.0316	0.09	6.90	**0.23**	**0.4026**	**0.01**
15	0.9112	0.9109	0.0344	**0.01**	8.56	0.94	0.6549	0.08
20	**0.9539**	0.9406	**0.0112**	0.79	8.53	1.48	0.8311	0.14

Table 6: **Cross-layer SAE comparison on the pooled `purified-PBMC` (in-distribution) and `PBMC` (out-of-distribution) datasets.** Statistics are computed over all gene-token rows in the combined train, validation, and test splits; no split of the external corpus was used to fit any SAE. Every configuration uses the same fixed TopK budget, so the mean L_0 is exactly 512 for all configurations. Bold marks the best value across layers within each dataset block. Layer 20 gives the strongest aggregate reconstruction on both datasets, while layer 10 has the most balanced feature dictionary.

E SAE Analysis and Visualizations

We analyze SAEs trained at different residual-stream layers to address two complementary questions. First, do the learned dictionaries support feature-level interpretation through broad coverage, selective activation, and distinct decoder directions? Second, how does feature organization vary with residual-stream depth? We assess reconstruction quality, feature-dictionary health, and performance relative to a dense linear baseline (PCA). Reconstruction metrics quantify how faithfully the SAE preserves the original activation vectors; feature-usage and decoder-geometry metrics identify inactive, overly common, or redundant dictionary elements; and the PCA comparison tests whether the SAE explains more variance than a fixed linear subspace. Detailed metric definitions and their usage are provided in Appendix C.1.

The cross-layer comparison uses epoch-10 checkpoints selected based on the ablation studies in Appendix D.2. We report *combined* results by pooling the train, validation, and test splits before computing each statistic. Table 6 presents results on the training-domain `purified-PBMC` dataset and the complete, disjoint `PBMC` dataset, which serves as an external out-of-distribution transfer test.

Layer depth reveals a tradeoff between reconstruction quality and feature robustness. No single layer performs best on every measure. Layer 20 reconstructs activations best overall on both datasets, explaining the most variance and preserving direction most faithfully. After row normalization reduces the influence of activation magnitude, however, layer 10 achieves slightly higher FVE. This reversal suggests that high-norm rows might contribute more to layer 20's aggregate advantage. Layer 15 ranks last on both aggregate and normalized metrics, making it the hardest activation space to reconstruct sparsely. These rankings persist on the disjoint `PBMC` dataset: layer 20 retains the best aggregate reconstruction, while layer 15 shows the largest loss. The pattern changes when feature use is considered. Layer 10 distributes its TopK assignments more evenly across the dictionary and maintains the most stable balance between selective and broadly active features. Layer 20 concentrates more assignments in frequently used features, and a small set of features active on the `purified-PBMC` dataset becomes inactive on `PBMC` dataset. Layer 15 again lies between the two. The preferred layer therefore depends on the goal: layer 20 is the better choice for preserving overall activation structure, whereas layer 10 combines stronger magnitude-controlled reconstruction with more balanced feature use. Viewed as a progression through model depth, these results are also consistent with features at layer 10 still taking shape and those at layer 20 being more fully consolidated, although the current metrics do not measure feature formation directly.

	Variance explained at 512 directions			Median PC distance	
Layer	SAE (%)	PCA (%)	Δ (pp)	SAE features	Random
10	97.59	89.71	+7.88	0.9187	0.9486
15	95.05	88.89	+6.16	0.9487	0.9488
20	97.62	95.77	+1.85	0.9480	0.9487

Table 7: **Reconstruction fidelity and directional alignment on `purified-PBMC` test split.** PC distance measures distance to the nearest of the leading 50 PCs using $1 - |\cos(\mathbf{d}, \mathbf{p}_i)|$. Random denotes the median distance for 4,096 isotropically sampled directions in the activation space. Larger distances indicate weaker alignment.

SAEs show broad feature coverage and distinct decoder directions. The dictionaries combine sparse activation with broad feature coverage. Despite the fixed sparse budget per token, feature use extends across nearly the entire dictionary. The feature-utilization curves (Figure 7) show that all features activate on `purified-PBMC` and almost all remain active under transfer. Most features activate infrequently, while universal features are rare. Activation rates generally decrease with depth, and the training and transfer distributions diverge modestly, suggesting greater selectivity and dataset sensitivity in deeper layers without widespread feature death. The decoder-redundancy distributions (Figure 8) show that regularly activated features have low cosine similarity to their nearest decoder neighbors, with near-duplicate directions accounting for only a small fraction of the dictionary. Under transfer to `PBMC`, activation frequencies span a broader range, particularly at deeper layers, but regularly used features continue to occupy distinct decoder directions. Together, these results support feature-level interpretation through broad coverage, selective activation, and low decoder redundancy.

SAEs capture more variance with weak alignment to PCA. On the `purified-PBMC` test split, SAEs preserve more activation variance than PCA at the trained budget of 512 directions per token (Table 7; Figure 9). The gains are consistent across all layers, indicating that the learned sparse representations retain information beyond that captured by the corresponding fixed linear subspace. This advantage decreases with depth. The comparison matches the number of directions used per token: PCA uses the same directions for every token, whereas the SAE selects token-specific features from its larger dictionary. Despite their reconstruction advantage, SAE decoder directions align only weakly with individual leading PCs. Median distances to the nearest of the leading 50 PCs are consistently over 0.9, indicating low absolute cosine similarities (Table 7). At layers 15 and 20, these medians are close to the random-direction baseline; layer 10 shows somewhat stronger alignment, although it remains weak in absolute terms. Together, these results show that SAEs retain more variance at the trained sparsity budget while learning directions that generally differ from individual leading PCs.

Taken together, these results answer both questions motivating our SAE analysis. Across both datasets, the dictionaries remain broadly utilized, predominantly selective, and almost entirely non-redundant, with these structural properties largely preserved under distribution shift. Residual-stream depth changes how this representational capacity is organized rather than uniformly improving SAE quality: deeper layers favor aggregate reconstruction but concentrate activation among a less balanced set of features, whereas earlier layers distribute feature use more evenly. On the training-domain test split, SAEs also consistently explain more variance than PCA at the trained sparsity budget, while their decoder directions align only weakly with individual leading PCs. The trained SAEs therefore satisfy the structural prerequisites for further feature-level analysis.

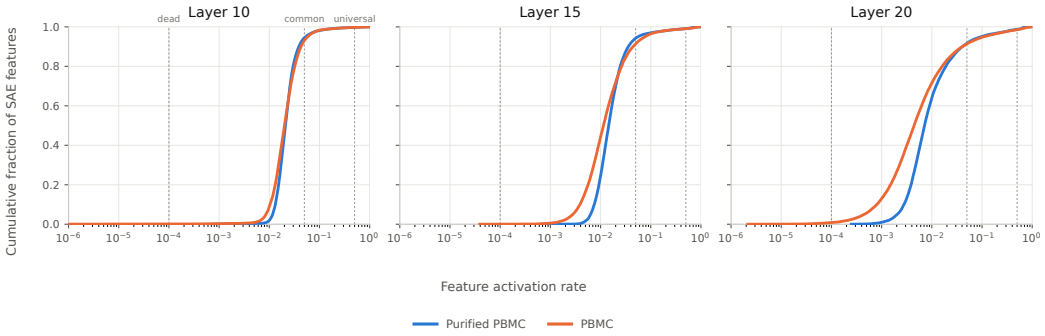


Figure 7: **Feature utilization across residual-stream layers.** Empirical cumulative distributions of feature activation rates for the `purified-PBMC` SAE dataset and the held-out `PBMC` dataset. Dotted lines mark the dead (10^{-4}), common (0.05), and universal (0.5) activation thresholds. At every layer, nearly all features activate above the dead threshold while most remain below the common threshold, demonstrating broad dictionary coverage with selective firing. From layers 10 to 20, the distributions broaden and shift toward lower activation rates, particularly on held-out `PBMC`, indicating increasingly selective and dataset-sensitive feature use without widespread feature death.

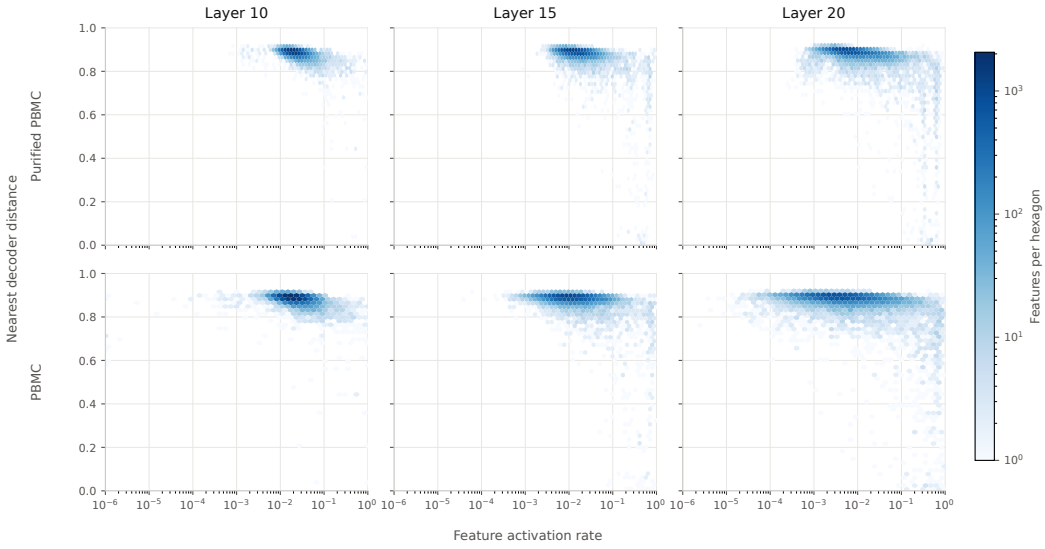


Figure 8: **Feature activation frequency and decoder distinctness.** Joint distributions of feature activation rate and nearest-decoder distance across layers and datasets; hexagon color denotes feature count on a shared logarithmic scale. Nearest-decoder distance is defined as $1 - |\cos(\mathbf{d}_i, \mathbf{d}_j)|$, so larger values indicate more distinct decoder directions. Most regularly activated features lie at distances roughly between 0.8 and 0.9, showing that frequent activation is distributed across distinct directions rather than duplicate decoder features. The held-out `PBMC` corpus exhibits a broader activation-frequency distribution than the `purified-PBMC` training corpus, especially at deeper layers, while retaining the same high-distance decoder structure.

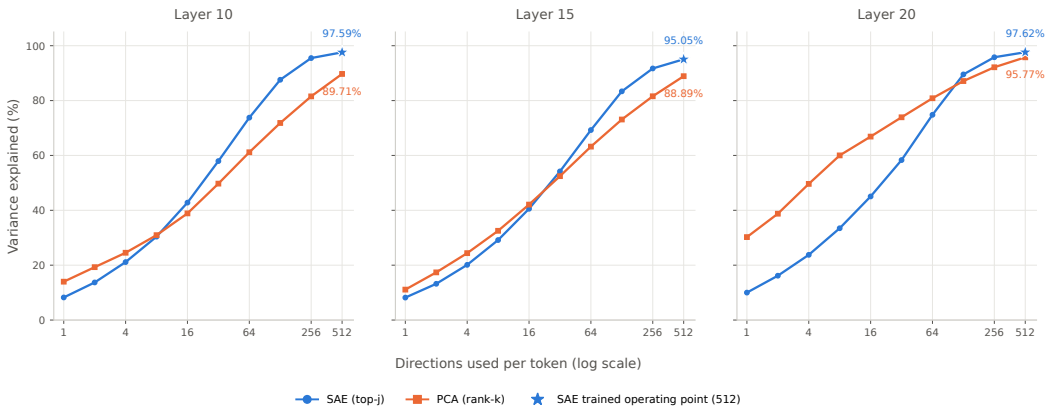


Figure 9: **Reconstruction fidelity of SAE features and PCA components.** SAEs and PCA are compared on the test split of the `purified-PBMC` dataset. Curves show the fraction of activation variance explained across all evaluated tokens as a function of the reconstruction budget, with a logarithmic horizontal axis. PCA reconstructs activations using a shared set of leading components, whereas each SAE selects decoder directions according to the largest feature activations for each token. The annotated percentages compare SAE and PCA reconstruction fidelity at the SAE's trained budget of 512 features per token. Note that the comparison matches the number of directions used per token, but not model capacity or parameter count.

F Pseudocode for Activation Editing Algorithm

Algorithm 1 Layer-ℓ activation editing for downstream evaluation. The hook is registered once and edits only the prefill pass of each prompt. Subsequent autoregressive decoding steps are not directly edited, although conditioned on the intervened representations.

Require: Base model $\mathcal{M}$; intervention layer ℓ; edit operator ϕ and intervention specification μ (Algorithm 2); evaluation split $\mathcal{D}$; sampling parameters $\pi = (T, \tau, p, k_{\text{gen}})$
Ensure: generation $y^{(c)}$ for each cell $c \in \mathcal{D}$

hook state: $\mathcal{I} \leftarrow \emptyset$; $S_{\text{prompt}} \leftarrow 0$; *armed* $\leftarrow$ FALSE
$\mathcal{M}' \leftarrow \textsc{Register}\big(\mathcal{M}, \ell, \textsc{Intervene}\big)$ ▷ once per run: every pass routes Layer$_\ell$ through Intervene
for all cells $c \in \mathcal{D}$ **do**
  $s \leftarrow (g_1, \ldots, g_N)$, the cell sentence of cell c
  $t \leftarrow t_{\text{pre}} \,\|\, \textsc{Tokenize}(s) \,\|\, t_{\text{post}}$; $S_{\text{prompt}} \leftarrow |t|$
  $(e_1, \ldots, e_N) \leftarrow \textsc{GeneEnds}(s)$; $\mathcal{I} \leftarrow \big\{\, |t_{\text{pre}}| + e_j - 1 \,\big\}_{j=1}^{N}$ ▷ final token of each gene
  armed $\leftarrow$ TRUE; $y^{(c)} \leftarrow \textsc{Generate}\big(\mathcal{M}', t; \pi\big)$; *armed* $\leftarrow$ FALSE; $\mathcal{I} \leftarrow \emptyset$
end for
return $\big(y^{(c)}\big)_{c \in \mathcal{D}}$

procedure $\textsc{Intervene}(H)$ ▷ $H \in \mathbb{R}^{B \times S \times d}$, the output of Layer$_\ell$
  if $\neg$ *armed* **or** $S \neq S_{\text{prompt}}$ **then**
    return H ▷ no-op; decoding steps have $S = 1$
  end if
  armed $\leftarrow$ FALSE ▷ consume the single prefill pass
  $Z \leftarrow H[:, \mathcal{I}, :]$; $H' \leftarrow H$; $H'[:, \mathcal{I}, :] \leftarrow \phi(Z, \mu)$ ▷ one batched call; other positions unchanged
  return H'
end procedure

Algorithm 2 Edit operators $\phi \in \{\textsc{Steer}, \textsc{ReconSteer}\}$ for the layer-ℓ hook of Algorithm 1. Both act on the gathered $Z \in \mathbb{R}^{B\times|\mathcal{I}|\times d}$ and return a tensor of the same shape using the shared latent edit Δ. Steer retains the original activation, whereas ReconSteer replaces it with the SAE reconstruction; thus, b_{dec} enters only through Dec_θ. Enc_θ includes the Top-k gate, and $W_{\text{dec}} \in \mathbb{R}^{d\times D}$ is the unit-norm dictionary. The intervention specification is $\mu = (q, \mathcal{F}_\downarrow, \mathcal{F}_\uparrow, \rho, a)$, where $q \in \{\textsc{Noop}, \textsc{Down}, \textsc{Up}, \textsc{Toward}\}$. Downward steering attenuates the latents in $\mathcal{F}_\downarrow$ by a fraction $\rho \in [0,1]$ of their observed magnitudes, with $\rho = 1$ corresponding to full ablation. Upward steering adds feature-specific magnitudes $a \in \mathbb{R}^D$ supported on $\mathcal{F}_\uparrow$. The Toward mode applies both edits simultaneously.

function $\textsc{Steer}(Z, \mu)$
  $z \leftarrow \text{Enc}_\theta(Z)$
  $\Delta \leftarrow \textsc{Edit}(z, \mu)$
  return $Z + \Delta W_{\text{dec}}^\top$
end function

function $\textsc{ReconSteer}(Z, \mu)$
  $z \leftarrow \text{Enc}_\theta(Z)$
  $\Delta \leftarrow \textsc{Edit}(z, \mu)$
  return $\text{Dec}_\theta(z + \Delta)$
end function

function $\textsc{Edit}(z, \mu)$
  $(q, \mathcal{F}_\downarrow, \mathcal{F}_\uparrow, \rho, a) \leftarrow \mu$
  $\Delta \leftarrow 0$
  if $q = \textsc{Noop}$ **then**
    return Δ
  end if
  if $q \in \{\textsc{Down}, \textsc{Toward}\}$ **then**
    $\Delta \leftarrow \Delta - \rho\big(z \odot \chi_{\mathcal{F}_\downarrow}\big)$
  end if
  if $q \in \{\textsc{Up}, \textsc{Toward}\}$ **then**
    $\Delta \leftarrow \Delta + \mathbf{1} \otimes a$
  end if
  return Δ
end function

G Cell Type Annotation and Error Analysis

G.1 Evaluation Setup and Failure Types

Because C2S models generate free-form text rather than labels from a closed set, we evaluate their predictions by matching the generated text against the three gold cell types, i.e. B cell, CD14 monocyte, and CD56 NK. The annotation template is provided in Prompt 1. An annotation is considered correct if it contains the gold cell type or an accepted synonym, with matching performed case-insensitively. Predictions satisfying this criterion are scored as correct; all others are scored as incorrect and assigned to one of the following error categories:

- **Under-specified**: The prediction is a valid ancestor of the gold label under the ontology's `is_a` hierarchy but is insufficiently specific. Examples include monocyte→"myeloid cell," B or NK cell→"lymphocyte," NK cell→"innate lymphoid cell," and any of the three gold types→"leukocyte."
- **Sibling**: The prediction identifies an incorrect terminal cell type within the correct lineage. Examples include monocyte→dendritic cell, macrophage, or neutrophil; NK cell→T cell; and B cell→T cell.
- **Wrong lineage**: The prediction crosses the myeloid–lymphoid divide. Examples include monocyte→B, NK, or T cell, and B or NK cell→monocyte or dendritic cell.
- **Off-target**: The prediction falls outside the set of mature cell types considered in the task. This category includes developmental antecedents, such as hematopoietic stem cells, common myeloid or lymphoid progenitors, precursor B cells, and generic "progenitor cells," as well as platelets, erythrocytes, non-hematopoietic cell types, and state-based labels such as "malignant cell." Developmental antecedents are not treated as under-specified predictions because they are distinct classes related to the gold type through `develops_from`, rather than superclasses of that type.
- **Vacuous**: The prediction conveys no meaningful cell-type information, as in "cell," "native cell," or "unknown." Although this category can be viewed as the limiting case of under-specification, we score it separately because it provides no useful information.

In the following sections, we report and compare error statistics for C2S Gemma-2-2B model, followed by a detailed analysis of their failure modes.

Prompt 1: Zero-shot cell-type annotation

```
The following is a list of gene names ordered by descending expression level
in a {organism} cell.  Your task is to give the cell type which this cell
belongs to based on its gene expression.
Cell sentence:  {cell_sentence}.
The cell type corresponding to these genes is:
```

G.2 Case Study: Diagnosis Workflow

We break down the annotation errors by cell types in Table 8. Across both datasets, CD14 monocytes are the most difficult to predict and account for most of the model's mistakes. We further classify these errors by failure mode in Table 9 and provide representative examples in Table 10.

Dataset	Run	Errors	B cell	CD14 monocyte	CD56 NK
Purified-PBMC	Base	211	7	172	32
	ℓ10	**200**	6	166	28
	ℓ15	210	6	172	32
	ℓ20	212	7	172	33
PBMC	Base	148	19	107	22
	ℓ10	**86**	19	56	11
	ℓ15	152	20	107	25
	ℓ20	147	19	106	22

Table 8: **Error counts on both datasets across all splits.**

Dataset	Run	Errors	Under-specified	Sibling	Wrong lineage	Off-target	Vacuous
Purified-PBMC	Base	211	5 (2.4)	148 (70.1)	35 (16.6)	19 (9.0)	4 (1.9)
	ℓ10	200	4 (2.0)	142 (71.0)	34 (17.0)	17 (8.5)	3 (1.5)
	ℓ15	210	6 (2.9)	147 (70.0)	35 (16.7)	18 (8.6)	4 (1.9)
	ℓ20	212	5 (2.4)	149 (70.3)	35 (16.5)	19 (9.0)	4 (1.9)
PBMC	Base	148	61 (41.2)	65 (43.9)	15 (10.1)	5 (3.4)	2 (1.4)
	ℓ10	86	24 (27.9)	40 (46.5)	15 (17.4)	5 (5.8)	2 (2.3)
	ℓ15	152	60 (39.5)	69 (45.4)	15 (9.9)	6 (3.9)	2 (1.3)
	ℓ20	147	59 (40.1)	66 (44.9)	15 (10.2)	5 (3.4)	2 (1.4)

Table 9: **The count of failure types for each run.** Each entry is count (row percentage %), note that the percentage may not sum to exactly 100 because of rounding.

Across datasets and layers, the errors are biologically structured rather than arbitrary. CD14 monocytes account for most errors and most of the improvement from layer 10, whereas B-cell errors are comparatively uncommon and remain stable across layers. In purified-PBMC, sibling confusions dominate at every layer, particularly CD14 monocytes mislabeled as dendritic cells and NK cells mislabeled as T cells. This pattern suggests that the model preserves broad lineage information but struggles to distinguish cell types with overlapping myeloid, antigen-presentation, or cytotoxic programs. PBMC exhibits the same near-lineage confusions but also elicits many under-specified predictions, most notably CD14 monocytes labeled simply as leukocyte. The layer-10 reconstruction alters this pattern most clearly: it reduces the two most prevalent failure modes—under-specification and sibling confusion—on both datasets, with particularly pronounced improvements on PBMC. By contrast, layers 15 and 20 remain close to the base model.

Failure mode	Gold label	Generated label	Total occurrences
Under-specified	B cell	"leukocyte"	8
	CD14 monocyte	"leukocyte"	190
	CD56 NK cell	"innate lymphoid cell"	6
Sibling	B cell	"naive thymus-derived CD4-positive, alpha-beta T cell"	24
	CD14 monocyte	"dendritic cell"	438
	CD56 NK cell	"CD4-positive, alpha-beta T cell"	102
Wrong lineage	B cell	"classical monocyte"	12
	CD14 monocyte	"CD4-positive, alpha-beta T cell"	56
	CD56 NK cell	"classical monocyte"	4
Off-target	B cell	"malignant cell"	27
	CD14 monocyte	"progenitor cell"	43
	CD56 NK cell	"hematopoietic stem cell"	12
Vacuous	B cell	"native cell"	8
	CD14 monocyte	"native cell"	15

Table 10: **Representative observed errors for C2S-Gemma-2-2B from each biological failure mode.** Note that occurrences are aggregated across all runs, including the base model and the models reconstructed at different layers, over all datasets and splits.

The representative outputs and their frequencies further illustrate how much biological information the model retains when it predicts incorrectly. Most errors preserve meaningful biological structure by identifying either a valid ancestor of the gold cell type or a closely related mature cell type. Wrong-lineage predictions indicate a more substantial loss of identity, while off-target and vacuous responses are the most severe failures because they provide little or no useful information about the mature cell type. Thus, the model generally fails by producing an insufficiently specific label or confusing biologically related cell types, rather than by losing cell-type information entirely.

To further investigate this prediction failure and assess our hypothesis, we analyzed layer 20 by comparing the mean feature activations of CD14 monocytes misclassified as dendritic cells with those of correctly classified CD14 monocytes in the `purified-PBMC` dataset. We selected the five most representative gained features and the five most representative lost features as predictions shifted from monocytes to dendritic cells. Each selected feature exhibited either a consistently high activation rate or a pronounced difference in activation rate between the two groups. Feature statistics are reported in Table 11, with the corresponding enrichment results presented in Table 12. The lost features consistently represent CD14-monocyte-associated signals, whereas most gained features are more closely associated with CD56 NK cells. Because our training data did not include specific dendritic-cell subtypes, the learned features may not capture the fine-grained signals required to distinguish dendritic cells from monocytes. Extending the analysis to finer-grained features across a broader range of cell classes could help identify the core dendritic-cell-specific signals underlying these prediction errors. Still, these results together reveal a coherent feature-level signature of misclassification: CD14-monocyte-associated features are lost, while features linked to alternative immune-cell identities, particularly CD56 NK cells, are gained. This structured shift suggests that the prediction errors reflect competing biological representations rather than arbitrary model behavior. More broadly, the correspondence between learned features and recognizable cell-type programs demonstrates the value of feature-based analysis for interpreting and diagnosing model predictions.

			Mean activation		Activation rate		
Feature	d	AUROC	Fail	Correct	Fail	Correct	Associated genes
F15265	-1.46	0.118	8.545	26.608	0.70	0.99	*MAZ*, *POLL*, *UCN*, *CRIPT*, *IMPACT*, *PTS*
F1146	-1.59	0.143	33.969	56.738	1.00	1.00	*CLEC7A*, *CLEC12A*, *CLEC4E*, *LINC00174*, *CYP27A1*, *CDKN2D*
F5465	-1.26	0.150	9.921	40.525	0.33	0.92	*RPS19BP1*, *COX7C*, *IGFBP2*, *HSP90AB1*, *COX6C*, *MPHOSPH10*
F3765	-0.97	0.185	3.084	10.503	0.47	0.90	*CLEC5A*, *CCL3L1*, *NINJ1*, *CXCL2*, *CXCL16*, *CCL3L3*
F14784	-1.03	0.200	3.999	11.169	0.54	0.90	*FER*, *TATDN2*, *PTCD2*, *GFER*, *TATDN3*, *PTPLAD2*
F12795	2.57	0.795	17.983	0.876	0.63	0.06	*RAB39A*, *RAB9A*, *RAB39B*, *RABGGTB*, *ZNF440*, *FAM8A1*
F8607	1.88	0.794	29.599	4.076	0.71	0.18	*FAM199X*, *RAB22A*, *ARL5A*, *FAM69A*, *FAM45A*, *ALDH1B1*
F3724	1.31	0.788	73.178	48.259	1.00	1.00	*GEMIN6*, *ORMDL3*, *FAM208A*, *SLC52A2*, *FAM8A1*, *FAM208B*
F3259	1.52	0.786	8.868	2.511	0.81	0.45	*DKFZP761J1410*, *CCDC13*, *MBNL2*, *MALSU1*, *TTC30A*, *TTC9*
F14059	1.21	0.768	17.324	8.666	0.96	0.92	*RGS18*, *RGS19*, *HIST1H2AC*, *RGS3*, *SIAH2*, *HPGD*

Table 11: **Selected layer-20 features lost or gained when CD14 monocytes are misclassified as dendritic cells.** Features in the *Lost in DC* group have lower activation in failed than in correct predictions ($d < 0$), whereas features in the *Gained in DC* group have higher activation in failed predictions ($d > 0$). AUROC measures separation between failed and correct predictions; values above 0.5 indicate higher activation in failed predictions.

H Natural Language Interpretation

Beyond generating cell sentences, C2S is reported to retain general language capabilities: given a cell, it can produce a natural-language description of that cell's state (Rizvi et al., 2026). Whereas cell-type annotation operates on individual cells, the natural-language interpretation task evaluates the model at the cluster level, testing whether it can reason over a group of cells and describe their shared biological properties. Prompt 2 gives the template used for this task. We randomly sample 20 clusters under each of the two-cell and three-cell settings (40 clusters in total) from the test split of `purified-PBMC` dataset for experiment and summarize our findings below.

Prompt 2: Natural Language Interpretation for Single/Multiple Cell(s)

```
Examine the top {num_genes} genes ordered by descending expression level in
{num_cells} {organism} {cell_noun}.  Based on this information, generate an
abstract summary which summarizes the biological insights contained in
{cell_reference}.
{cells}
Abstract summary:
```

Generations 1 and 2 show representative outputs. Across random seeds and cell-type variants, the generated text remained highly similar, repeatedly referring to SARS-CoV-2 infection and control conditions or COVID-19 control samples without providing cell-specific interpretations. These results suggest that C2S Gemma-2-2B does not reliably produce cell-specific natural-language interpretations under the evaluated conditions. We therefore exclude free-form language generation tasks from further evaluation and focus on cell-type annotation.

Feature	Representative GO term	p-value	Specificity interpretation
Lost Features			
F15265	Extracellular exosome	1.487×10^{-4}	Extracellular and secretory program compatible with innate myeloid activity
	Humoral immune response	1.493×10^{-3}	Broad immune signal supporting the monocyte interpretation
F1146	Secondary active transmembrane transporter activity	4.720×10^{-25}	Strong ion and solute transport program; physiological rather than direct monocyte identity
F5465	Cytochrome complex	2.048×10^{-11}	Mitochondrial metabolic-state signal rather than cell-type identity
	Mitochondrial electron transport, cytochrome c to oxygen	3.112×10^{-8}	Oxidative-energy program compatible with metabolically active monocytes
F3765	Cellular response to chemical stimulus	2.235×10^{-8}	Activated innate-cell response program
	Cell migration	5.568×10^{-8}	Migratory and recruitment-related program consistent with monocyte behavior
F14784	MAP kinase phosphatase activity	1.067×10^{-5}	Immune-signaling regulatory program, although not lineage-specific
Gained Features			
F12795	Zinc ion binding	9.950×10^{-109}	Extremely broad regulatory enrichment
	DNA-binding transcription factor activity	7.185×10^{-94}	General transcriptional regulation
F8607	GTPase activity	9.694×10^{-15}	Intracellular signaling and trafficking program
	Vesicle-mediated transport	8.041×10^{-10}	Potentially compatible with immune-granule trafficking
F3724	Mitochondrial large ribosomal subunit	7.440×10^{-14}	Mitochondrial housekeeping program; weakly B-cell-associated
F3259	RNA splicing	6.146×10^{-5}	Generic nuclear RNA-processing program
F14059	Molecular function regulator activity	3.051×10^{-9}	Broad regulatory signal
	Signal transduction	1.982×10^{-3}	Underlying genes include `CLEC4A`, `FCER1A`, and `CSF1R`, suggesting a weak cDC2-like fragment

Table 12: **Representative GO enrichments for features in Table 11.** The first five form a comparatively coherent CD14-monocyte program involving secretion, transport, mitochondrial metabolism, migration, and signaling. The last five are mostly associated with CD56 NK cells through differential feature activation, but their GO enrichments are dominated by generic transcriptional, trafficking, mitochondrial, and RNA-processing programs. However, only F14059 contains a weak cDC2-like marker fragment, and no feature shows a complete dendritic-cell or antigen-presentation program.

Generation 1 — Natural Language Interpretation (two cells)

Dataset	`purified-PBMC`
Model	`Cell-to-Sentence Gemma-2-2B`
Cell Names	{103295, 98084}
Gold labels	`{cd14_monocytes, cd56_nk}`
Decoding	`top_k` = 30, `top_p` = 0.9, `max_new_tokens` = 1024
Seed	61

Baseline

This study created a reusable datamine to compare single-cell transcriptome profiles of SARS-CoV-2 infection. By harmonizing datasets from COVID-19 and control conditions, the researchers derived gene signature modules for different cell types and clinical conditions. They developed an interactive web portal (ToppCell) and proposed three hypotheses based on their findings: multicellular signaling among monocyte-derived macrophages, novel platelet subtypes, and a cell activator network driving B cell maturation..

Layer 10 Reconstructed

This study created a database called ToppCell to explore single-cell transcriptome profiles of SARS-CoV-2 infection and control conditions. They identified gene signature modules for different cell types and clinical conditions. Using ToppCell, they proposed hypotheses about multicellular signaling among macrophages, novel platelet subtypes, and a cell activator network driving B cell maturation. The findings are accessible through an interactive web portal..

Layer 15 Reconstructed

This study created a reusable datamine to compare single-cell transcriptome profiles of SARS-CoV-2 infection. By harmonizing datasets from COVID-19 and control conditions, the researchers derived gene signature modules for different cell types and clinical conditions. They developed an interactive web portal (ToppCell) and proposed three hypotheses based on their findings: multicellular signaling among monocyte-derived macrophages, novel platelet subtypes, and a cell activator network driving B cell maturation..

Layer 20 Reconstructed

This study created a database called ToppCell to explore single-cell transcriptome profiles of SARS-CoV-2 infection and control conditions. They identified gene signature modules for different cell types and clinical conditions. Using ToppCell, they proposed hypotheses about multicellular signaling among macrophages, novel platelet subtypes, and a cell activator network driving B cell maturation. The findings are accessible through an interactive web portal..

Generation 2 — Natural Language Interpretation (three cells)

Dataset	`purified-PBMC`
Model	`Cell-to-Sentence Gemma-2-2B`
Cell Names	{99195, 105631, 66262}
Gold labels	`{cd56_nk, cd14_monocytes, b_cells}`
Decoding	`top_k` = 30, `top_p` = 0.9, `max_new_tokens` = 1024
Seed	58

Baseline

This study created a database called ToppCell to explore single-cell gene expression data from COVID-19 and control samples. They identified gene signature modules for different cell types and clinical conditions. Using ToppCell, they proposed hypotheses including a signaling cascade in macrophages for T cell recruitment, novel platelet subtypes related to coagulation, and a cell activator network linked to autoimmunity risk..

Layer 10 Reconstructed

This study created a database called ToppCell to explore single-cell gene expression data from COVID-19 and control samples. They identified gene signature modules for different cell types and clinical conditions. Using ToppCell, they proposed hypotheses including a signaling cascade in macrophages for T cell recruitment, novel platelet subtypes related to coagulation, and a cell activator network linked to autoimmunity risk..

Layer 15 Reconstructed

This study created a database called ToppCell to explore single-cell gene expression data from COVID-19 and control samples. They identified gene signature modules for different cell types and clinical conditions. Using ToppCell, they proposed hypotheses including a signaling cascade in macrophages for T cell recruitment, novel platelet subtypes related to coagulation, and a cell activator network linked to autoimmunity risk..

Layer 20 Reconstructed

This study created a database called ToppCell to explore single-cell gene expression data from COVID-19 and control samples. They identified gene signature modules for different cell types and clinical conditions. Using ToppCell, they proposed hypotheses including a signaling cascade in macrophages for T cell recruitment, novel platelet subtypes related to coagulation, and a cell activator network linked to autoimmunity risk..

I Feature Comprehension and Enrichment Analysis

As depth increases, the selected features appear more consistently organized by cell type, whereas earlier-layer features are more dispersed and overlapping. This pattern suggests increasingly coherent cell-type-associated directions in deeper layers. However, UMAP distorts distances and each layer's features were selected independently, so the apparent consolidation should be verified in the original feature space.

This section provides supplemental artifacts supporting the feature interpretations in Section 4. Figure 10 presents the top 15 features for each cell type and their corresponding mean activation magnitudes across all selected layers. We selected the top 15 features per population because specificity margins were already small by rank 15 at layers 10 and 15, and applied the same cutoff at layer 20 for consistency.

Table 13 identifies marker-bearing features within the same top-feature set and reports their recovered marker genes, specificity ranks, activation rates, and one-vs-rest separation metrics. Tables 14, 15, and 16 report representative immune-function-related enrichment terms associated with each feature; asterisks indicate terms directly related to the corresponding cell type.

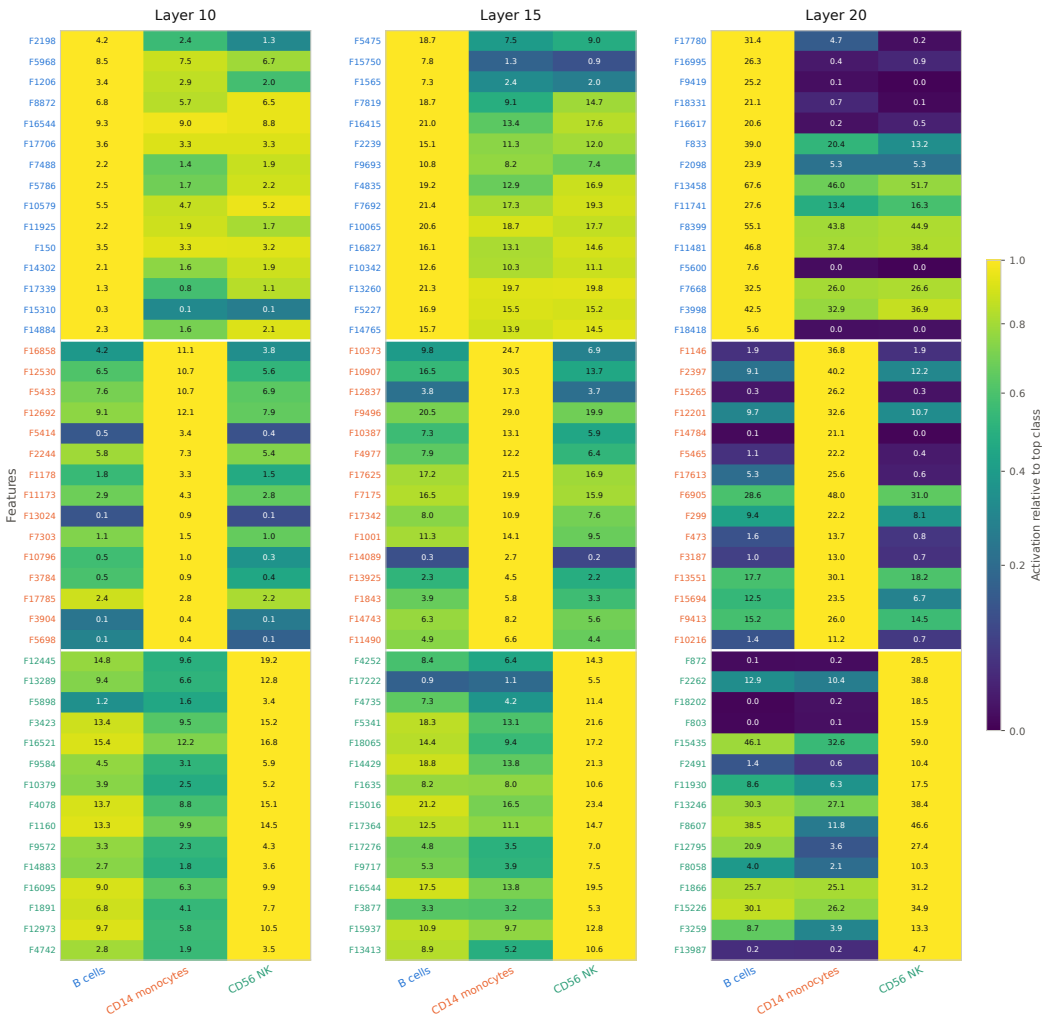


Figure 10: **Cell-type-selective SAE features measured on `purified-PBMC` dataset.** Mean activation of the most class-selective SAE features at layers $\ell \in \{10, 15, 20\}$. Each panel shows the 15 highest-specificity-margin features per class (rows, 45 in total), evaluated against all three cell classes (columns). Row labels give the SAE feature index, colored by the class the feature was ranked for; white rules separate the three per-class blocks. Cell color shows each feature's mean activation divided by its largest class mean; printed values are the raw means. By construction every feature peaks on its target class (yellow diagonal blocks).

Layer	Feature	Rank	Margin	Act. rate	AUROC	d	Recovered marker genes | MHC-II
B cells							
10	**F2198**	1	1.9	0.78	0.64	0.53	*TNFRSF13C*, *LINC00926*, *TNFRSF13B*, *CD19*, *CD22*, *ARHGAP24*, *FCRL5*
15	**F15750**	2	6.4	0.93	0.89	1.43	*CD22*, *CD19*, *LINC00926*, *CD79A*, *TNFRSF13C*, *CD79B*, *ARHGAP24* | *CD74*
	F1565	3	4.9	0.88	0.77	1.03	*LINC00926*, *CD22*, *CD19*, *TNFRSF13C*, *TNFRSF13B* | *CD74*
20	**F17780**	1	26.7	1.00	0.99	3.12	*RALGPS2*, *CD79B*, *CD79A* | *HLA-DQA1*, *HLA-DQA2*, *HLA-DQB1*, *CD74*
	F16995	2	25.4	0.98	0.99	3.24	*LINC00926*
	F9419	3	25.0	0.92	0.96	2.63	*LINC00926*, *CD22*, *CD79B* | *HLA-DQA1*, *HLA-DQA2*, *HLA-DMB*, *HLA-DOB*, *CD74*
	F18331	4	20.4	0.96	0.97	2.66	*RALGPS2*, *LINC00926*, *FOXP1* | *HLA-DQA2*
	F16617	5	20.1	0.99	1.00	4.74	*CD79B*, *RALGPS2*, *CD79A*
	F5600	12	7.5	0.76	0.88	1.96	*CD79B*, *CD79A*, *EBF1*
CD14 monocytes							
10	**F5414**	5	2.9	0.68	0.75	0.99	*SLC7A7*, *PLAUR*, *SLC11A1*, *CLEC4A*, *CLEC4E*, *CLEC7A*
15	**F14089**	11	2.4	0.44	0.68	0.83	*S100A8*, *S100A12*, *S100A9*, *TNFAIP2*, *MNDA*
	F1843	13	2.0	0.86	0.66	0.65	*CLEC9A*, *CLEC10A* | *HLA-DPB1*, *HLA-DPA1*
20	**F1146**	1	34.9	1.00	1.00	5.01	*CLEC7A*, *CLEC12A*, *CLEC4E*, *SLC11A1*, *CSF3R*
	F2397	2	28.0	0.99	0.92	1.93	*CSF3R*, *S100A12*, *CLEC4E*, *CLEC12A*, *CLEC7A*
	F14784	5	21.1	0.97	0.99	4.34	*CLEC7A*
	F5465	6	21.1	0.77	0.87	1.91	*SLC11A1*
	F17613	7	20.3	0.98	0.96	2.62	*CXCL16*, *SLC11A1*, *CD68*, *CLEC7A*
	F15694	13	10.9	0.98	0.84	1.31	*CLEC9A*, *CLEC10A* | *HLA-DPB1*, *HLA-DPA1*, *HLA-DOA*, *HLA-DQB1*, *HLA-DRB1*
CD56 NK							
10	**F5898**	3	1.9	0.82	0.67	0.74	*KIR2DL3*, *CCL5*, *KLRD1*, *KLRB1*, *SH2D1B*, *GZMH*, *KLRG1*, *CD160*
15	**F17222**	2	4.4	0.68	0.72	0.78	*LINC00861*, *GNLY*, *CCL4*, *CCL5*, *CST7*, *NKG7*
	F3877	13	1.9	0.83	0.61	0.50	*GNLY*
20	**F872**	1	28.3	0.99	0.99	3.75	*KLRD1*, *KLRF1*, *GNLY*, *CCL5*
	F2262	2	25.9	0.98	0.91	2.14	*CCL5*, *CX3CR1*, *GNLY*
	F803	4	15.8	0.91	0.95	2.46	*GNLY*, *KLRF1*, *KLRD1*
	F13987	15	4.5	0.48	0.72	1.19	*GZMH*

Table 13: **Marker-bearing features among the top 15 per cell class and the marker genes recovered from the ten highest-activation genes of each feature.** The marker genes are ordered by activation and listed in full; other genes are not shown. MHC-II genes are reported after the bar because they are shared by B cells and antigen-presenting myeloid cells. Rank is within each class by specificity margin, and act. rate represents the fraction of cells in that class in which the feature is active. AUROC and Cohen's d are one-vs-rest separations computed on the feature activation between cells of that class and all remaining cells.

Feature	Representative GO term	Adjusted p	Specificity interpretation
B cells			
F2198	Signaling receptor activity	4.585×10^{-36}	General cell-surface receptor signaling
	Lymphocyte activation	8.407×10^{-26}	Broad lymphoid signal
	Leukocyte activation	1.982×10^{-23}	Broad immune-activation signal
F14884	NOD1 signaling pathway	4.533×10^{-3}	Shared innate pattern-recognition pathway
	Toll-like receptor 4 signaling pathway	1.472×10^{-2}	Innate immune signaling
CD56 NK			
F5898	Immune system process	2.447×10^{-15}	Broad immune-system signal
	Immune response	6.631×10^{-15}	General immune-response signal
	Leukocyte activation	5.778×10^{-11}	Broad immune activation
F16095	Lymphoid progenitor cell differentiation	3.163×10^{-2}	Broad lymphoid development; not mature NK-cell identity
CD14 monocytes			
F5414	Antifungal innate immune response	4.599×10^{-7}	Innate myeloid-associated function
	Pattern recognition receptor activity	9.162×10^{-6}	Shared innate immune sensing
	Innate immune response	9.280×10^{-4}	Broad innate-immune signal
F10796	Endocytic vesicle membrane	7.833×10^{-6}	General endocytic and vesicular signal
	Antigen processing and presentation of endogenous antigen	5.520×10^{-4}	Shared antigen-presentation signal
	Phagocytic vesicle membrane	1.478×10^{-3}	Phagocyte-aligned function; not CD14-specific

Table 14: **Representative GO enrichments at layer 10.** Most terms describe broad immune or cellular functions rather than direct cell-type identity.

Feature	Representative GO term	Adjusted p	Specificity interpretation
B cells			
F15750	MHC class II protein complex	2.986×10^{-8}	Shared antigen-presentation machinery; B-cell-aligned with associated markers
	Antigen processing and presentation of exogenous antigen	1.628×10^{-6}	Shared antigen-presentation signal
	Adaptive immune response	1.170×10^{-5}	Broad adaptive-immune signal
F1565	Adaptive immune response	3.955×10^{-7}	Broad adaptive-immune signal
	Leukocyte activation	3.955×10^{-7}	Broad immune-activation signal
CD56 NK			
F17222	Regulation of natural killer cell chemotaxis	8.440×10^{-4}	NK-directed migration program*
	Natural killer cell chemotaxis	2.148×10^{-3}	NK-directed migration program*
	Immune response	2.294×10^{-3}	General immune-response signal
F9717	Inhibitory MHC class I receptor activity	6.788×10^{-4}	Immune-receptor function
	MHC class I receptor activity	3.956×10^{-3}	Shared immune-receptor function
CD14 monocytes			
F14089	Calprotectin complex	2.143×10^{-4}	S100A8/A9-associated myeloid program shared with granulocytes
	RAGE receptor binding	4.281×10^{-5}	Shared inflammatory signaling function
	Toll-like receptor 4 binding	5.907×10^{-4}	Shared innate inflammatory signaling function
F12837	Myeloid leukocyte migration	1.251×10^{-5}	Myeloid-aligned trafficking
	Leukocyte migration	1.251×10^{-5}	Broad immune-cell trafficking signal

Table 15: **Representative GO enrichments at layer 15.** NK-associated chemotaxis and myeloid-associated calprotectin and migration accompany shared immune programs.

Feature	Representative GO term	Adjusted p	Specificity interpretation
B cells			
F5600	IgM B-cell receptor complex	1.625×10^{-3}	B-cell receptor machinery*
	B-cell receptor complex	5.819×10^{-3}	B-cell receptor machinery*
	Immunoglobulin complex	1.443×10^{-2}	B-lineage-associated immune signal
F17780	Adaptive immune response	6.263×10^{-14}	Broad adaptive-immune signal
	MHC class II protein complex	3.627×10^{-13}	Shared antigen-presentation signal
CD56 NK			
F872	Natural killer cell-mediated immunity	3.982×10^{-2}	NK-lineage-related immune function*
F2491	Lymphocyte activation	2.962×10^{-6}	Broad lymphoid signal
	Mononuclear cell differentiation	9.371×10^{-6}	Broad immune-lineage signal
	Leukocyte activation	9.371×10^{-6}	Broad immune-activation signal
CD14 monocytes			
F15694	MHC protein complex	2.787×10^{-13}	Shared antigen-presentation signal
	MHC class II protein complex	3.888×10^{-9}	Shared antigen-presentation signal
	Antigen processing and presentation of peptide antigen	6.199×10^{-9}	Shared antigen-presentation signal
F15265	Extracellular exosome	1.487×10^{-4}	General extracellular/secretory signal
	Humoral immune response	1.493×10^{-3}	Broad immune signal

Table 16: **Representative GO enrichments at layer 20.** B/NK-associated terms accompany shared antigen-presentation and immune programs.

J Steering Effect and Analysis

J.1 Steer Modes Comparison

We evaluate intervention effectiveness across the three steering modes described in Algorithm 2: *steer up*, *steer down*, and *steer toward*. For feature amplification, we use additive magnitudes in the range $[20, 50]$, with the upper bound corresponding to the highest feature activation magnitude shown in Figure 10. For feature suppression, we reduce the selected feature activations by 99%. Notably, these operations differ in their dependence on the initial feature activations. Amplification adds the selected feature directions to the original or reconstructed embedding regardless of whether those features are initially active, while retaining existing feature contributions. Suppression, by contrast, affects only features that are already active; selecting an inactive feature for suppression will leave the embedding unchanged.

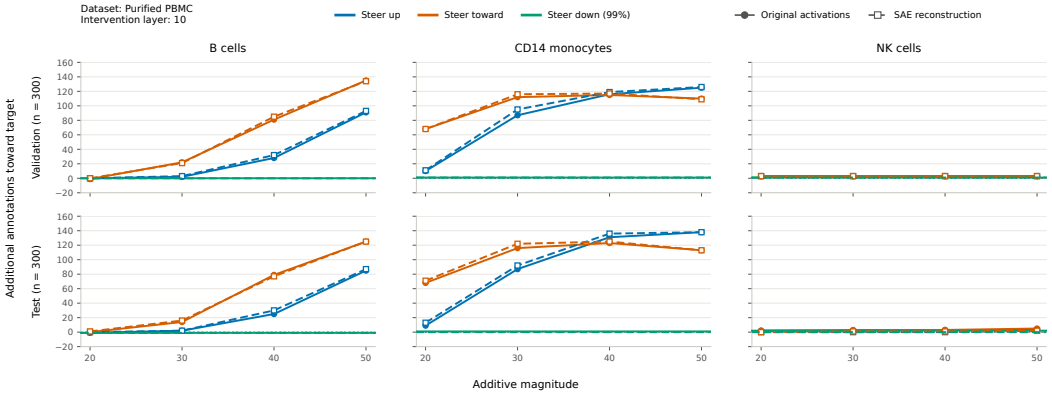


Figure 11: **Impact of different steering methods on intervening cell-type annotation at layer 10.** *Steer up* strengthens 15 target-associated features; *steer down* suppresses 30 selected features associated with the other cell types by 99%; *steer toward* combines both operations.

Figure 11 shows that suppressing features associated with non-target cell types has little effect on steering annotations toward the target label. Because 99% suppression already nearly eliminates these feature activations without producing a meaningful effect, we did not evaluate additional suppression rates. In contrast, amplifying features representative of the target cell type substantially improves steering effectiveness. *Steer toward*, which combines target-feature amplification with non-target-feature suppression, achieves the strongest performance for B cells across all tested amplification magnitudes and also performs well for CD14 monocytes. These results support retaining *steer toward* as the preferred steering mode among those evaluated. Moreover, an amplification magnitude of 20 consistently underperforms the larger tested magnitudes, suggesting that stronger amplification is more effective within the evaluated range. Additionally, we evaluated steering mode and amplification magnitude on the `PBMC` dataset at layer 10 (Figure 12), observing broadly consistent trends.

J.2 Cross-Layer Steering Investigation

In this section, we examine cross-layer feature steering and its effects on cell-type annotations. Table 17 reports both the total number of annotation changes and the number of transitions into the target class. Features from source layer 15 induce the most annotation changes. However, for B-cell steering, most changes are to the malignant-cell label rather than the intended B-cell label. Conversely, features from source layer 20 more effectively steer predictions toward the B-cell label, but are less effective for NK-cell steering than features from layer 15.

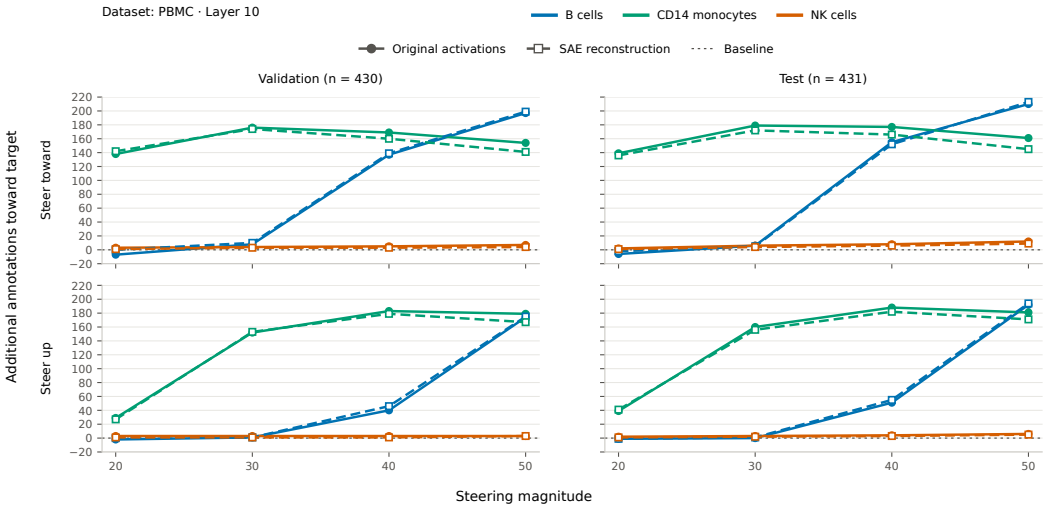


Figure 12: **Layer 10 steering across magnitudes on `PBMC` dataset.** This figure complements Figures 4 and 11.

			Source layer 10		Source layer 15		Source layer 20	
Dataset	Target	Magnitude	Val.	Test	Val.	Test	Val.	Test
`purified-PBMC`	B cells	30	47 (2)	56 (2)	243 (57)	260 (57)	162 (124)	151 (117)
		40	77 (28)	77 (25)	236 (64)	252 (66)	175 (144)	165 (134)
		50	105 (91)	96 (85)	220 (80)	223 (79)	165 (121)	150 (109)
	CD14 monocytes	30	93 (87)	99 (87)	215 (207)	220 (218)	173 (131)	187 (138)
		40	128 (116)	148 (131)	215 (211)	223 (221)	207 (194)	222 (199)
		50	139 (125)	157 (138)	215 (215)	223 (221)	208 (191)	222 (198)
	NK cells	30	4 (3)	5 (2)	207 (207)	208 (208)	132 (34)	153 (37)
		40	5 (3)	5 (3)	207 (207)	208 (208)	144 (64)	160 (69)
		50	6 (3)	5 (3)	207 (207)	208 (208)	136 (67)	153 (68)
`PBMC`	B cells	30	61 (1)	66 (1)	314 (135)	322 (163)	213 (103)	235 (122)
		40	118 (40)	138 (51)	264 (227)	274 (242)	211 (87)	234 (94)
		50	196 (174)	211 (192)	250 (245)	268 (262)	172 (44)	175 (40)
	CD14 monocytes	30	176 (152)	169 (160)	236 (216)	225 (210)	240 (25)	235 (28)
		40	209 (183)	205 (188)	237 (235)	225 (223)	240 (47)	230 (45)
		50	212 (179)	205 (181)	237 (235)	225 (224)	234 (68)	227 (72)
	NK cells	30	8 (3)	11 (3)	382 (382)	383 (383)	180 (71)	167 (62)
		40	10 (3)	13 (4)	382 (382)	383 (383)	196 (122)	193 (121)
		50	12 (3)	17 (6)	382 (382)	383 (383)	194 (131)	186 (120)

Table 17: **Prediction changes induced by additive up-steering at layer 10 using features from different source layers.** Each entry reports the total number of changed predictions, with changes into the target class in parentheses, separately for validation and test. Changes are measured against the unsteered model.

We therefore hypothesize that **stronger steering effects are associated with clearer connections between learned features and cell-type-specific biological concepts**, as reflected by greater enrichment for the corresponding biological processes, i.e., lower adjusted GO-enrichment p-values. As shown in Tables 15 and 16, enrichment for B-cell-specific biological processes becomes more apparent at layer 20, consistent with the weaker target-directed steering observed using features from earlier layers. For CD14 monocytes, layer-15 features are enriched for myeloid-related processes, whereas layer-20 features predominantly reflect broader antigen-related processes, potentially contributing to reduced steering effectiveness. For NK cells, features from both layers support steering, but those from layer 15 exhibit lower adjusted GO-enrichment p-values, consistent with their stronger steering effects.

Together, these observations support the hypothesis that clearer alignment with cell-type-specific biological concepts is associated with more effective target-directed steering.

Limitations

Our study has several limitations that define its scope. First, we analyze only C2S Gemma-2-2B; applying the pipeline to the 27B variant exceeded our computational budget, and larger models may organize biological information differently across layers. Second, as a pilot study, we deliberately restrict our analysis to three well-separated PBMC populations with high-confidence labels, which allows a controlled test of concept alignment but cannot resolve finer distinctions, such as dendritic-cell subtypes. Third, we use a single SAE family (TopK); hierarchical architectures such as Matryoshka SAEs may better capture the broad-to-specific structure of biological concepts. Finally, the annotation downstream task uses greedy decoding and is therefore deterministic, but each SAE is trained with a single seed, so individual features may vary across runs; we accordingly emphasize layer-wise trends over individual features. Extending the framework to larger models, more diverse data, and alternative SAE architectures is left to future work.

AI Declaration

We acknowledge the use of GPT-6 Astra (OpenAI) to assist with writing and language polishing, the discovery of related work, and portions of the code implementation. Claude 5 Opus (Anthropic) was used to assist with figure and table preparation. The authors take full responsibility for the final content, analyses, and conclusions.

References

10x Genomics. 4k PBMCs from a Healthy Donor. `https://www.10xgenomics.com/datasets/4-k-pbm-cs-from-a-healthy-donor-2-standard-2-1-0`, 2017a.

10x Genomics. 8k PBMCs from a Healthy Donor. `https://www.10xgenomics.com/datasets/8-k-pbm-cs-from-a-healthy-donor-2-standard-2-1-0`, 2017b.

Etowah Adams, Liam Bai, Minji Lee, Yiyang Yu, and Mohammed AlQuraishi. From mechanistic interpretability to mechanistic biology: training, evaluating, and interpreting sparse autoencoders on protein language models. In *Proceedings of the 42nd International Conference on Machine Learning*, ICML'25. JMLR.org, 2025.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nicholas L. Turner, Cem Anil, Carson Denison, Amanda Askell, et al. Towards monosemanticity: decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023. https://transformer-circuits.pub/2023/monosemantic-features.

Yiqun Chen and James Zou. Genept: a simple but effective foundation model for genes and cells built from chatgpt. *BioRxiv*, pp. 2023–10, 2024.

Charlotte Claye, Pierre Marschall, Wassila Ouerdane, CELINE HUDELOT, and Julien Duquesne. A framework to extract and interpret biological concepts from scRNAseq generative foundation models. In *ICML 2025 Generative AI and Biology (GenBio) Workshop*, 2025. URL `https://openreview.net/forum?id=wZpxbCwEe4`.

Haotian Cui, Chloe Wang, Hassaan Maan, Kuan Pang, Fengning Luo, Nan Duan, and Bo Wang. scgpt: toward building a foundation model for single-cell multi-omics using generative ai. *Nature methods*, 21(8):1470–1480, 2024.

Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models, 2023. URL `https://arxiv.org/abs/2309.08600`.

Joshua Engels, Logan Riggs Smith, and Max Tegmark. Decomposing the dark matter of sparse autoencoders. *ArXiv*, abs/2410.14670, 2024. URL `https://api.semanticscholar.org/CorpusID:273482303`.

Yin Fang, Xinle Deng, Kangwei Liu, Ningyu Zhang, Jingyang Qian, Penghui Yang, Xiaohui Fan, and Huajun Chen. A multi-modal ai copilot for single-cell analysis with instruction following. *arXiv preprint arXiv:2501.08187*, 2025.

Jack Gallifant, Shan Chen, Kuleen Sasse, Hugo Aerts, Thomas Hartvigsen, and Danielle Bitterman. Sparse autoencoder features for classifications and transferability. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 29939–29963, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.1521. URL https://aclanthology.org/2025.emnlp-main.1521/.

Leo Gao, Tom Dupre la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. Scaling and evaluating sparse autoencoders. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=tcsZt9ZNKD.

Asma Ghandeharioun, Ann Yuan, Marius Guerard, Emily Reif, Michael A. Lepori, and Lucas Dixon. Who's asking? user personas and the mechanics of latent misalignment. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 125967–126003. Curran Associates, Inc., 2024. doi: 10.52202/079017-4002. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/e40d5118ee8f837729fa877add71c38f-Paper-Conference.pdf.

Minsheng Hao, Jing Gong, Xin Zeng, Chiming Liu, Yucheng Guo, Xingyi Cheng, Taifeng Wang, Jianzhu Ma, Xuegong Zhang, and Le Song. Large-scale foundation model on single-cell transcriptomics. *Nature methods*, 21(8):1481–1491, 2024.

Yuhan Hao, Stephanie Hao, Erica Andersen-Nissen, William M Mauck, Shiwei Zheng, Andrew Butler, Maddie J Lee, Aaron J Wilk, Charlotte Darby, Michael Zager, et al. Integrated analysis of multimodal single-cell data. *Cell*, 184(13):3573–3587, 2021.

Liis Kolberg, Uku Raudvere, Ivan Kuzmin, Priit Adler, Jaak Vilo, and Hedi Peterson. g: Profiler—interoperable web service for functional enrichment analysis and gene identifier mapping (2023 update). *Nucleic acids research*, 51(W1):W207–W212, 2023.

Taku Kudo and John Richardson. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Conference on Empirical Methods in Natural Language Processing*, 2018. URL https://api.semanticscholar.org/CorpusID:52051958.

Tom Lieberum, Senthooran Rajamanoharan, Arthur Conmy, Lewis Smith, Nicolas Sonnerat, Vikrant Varma, Janos Kramar, Anca Dragan, Rohin Shah, and Neel Nanda. Gemma scope: Open sparse autoencoders everywhere all at once on gemma 2. In Yonatan Belinkov, Najoung Kim, Jaap Jumelet, Hosein Mohebbi, Aaron Mueller, and Hanjie Chen (eds.), *Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pp. 278–300, Miami, Florida, US, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.blackboxnlp-1.19. URL https://aclanthology.org/2024.blackboxnlp-1.19/.

Daniel Morales-Brotons, Thijs Vogels, and Hadrien Hendrikx. Exponential moving average of weights in deep learning: Dynamics and benefits. *Trans. Mach. Learn. Res.*, 2024, 2024. URL https://api.semanticscholar.org/CorpusID:270355881.

Flavia Pedrocchi, Florian Barkmann, Amir Joudaki, and Valentina Boeva. Sparse autoencoders reveal interpretable features in single-cell foundation models. *bioRxiv*, 2025. doi: 10.1101/2025.10.22.681631. URL https://www.biorxiv.org/content/early/2025/10/23/2025.10.22.681631.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Turner. Steering llama 2 via contrastive activation addition. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15504–15522, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.828. URL https://aclanthology.org/2024.acl-long.828/.

Syed Asad Rizvi, Daniel Levine, Aakash Patel, Shiyang Zhang, Eric Wang, Curtis Jamison Perry, Ivan Vrkic, Nicole Mayerli Constante, Zirui Fu, Sizhuang He, et al. Scaling large language models for next-generation single-cell analysis. *BioRxiv*, pp. 2025–04, 2026.

Viktoria Schuster. Can sparse autoencoders make sense of gene expression latent variable models? In David A Knowles and Peter K Koo (eds.), *Proceedings of the 20th Machine Learning in Computational Biology meeting*, volume 311 of *Proceedings of Machine Learning Research*, pp. 81–94. PMLR, 10–11 Sep 2025. URL https://proceedings.mlr.press/v311/schuster25a.html.

Yaorui Shi, Jiaqi Yang, Changhao Nai, Sihang Li, Junfeng Fang, Xiang Wang, Zhiyuan Liu, and Yang Zhang. Language-enhanced representation learning for single-cell transcriptomics. *arXiv preprint arXiv:2503.09427*, 2025.

Elana Simon and James Zou. Interplm: discovering interpretable features in protein language models via sparse autoencoders. *Nature methods*, 22(10):2107–2117, 2025.

Xiaoqing Sun, Alessandro Stolfo, Joshua Engels, Ben Wu, Senthooran Rajamanoharan, Mrinmaya Sachan, and Max Tegmark. Dense sae latents are features, not bugs. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, pp. 140252–140282. Curran Associates, Inc., 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/cd091a4d8e97157d32940428f902c7b0-Paper-Conference.pdf.

Christina V Theodoris, Ling Xiao, Anant Chopra, Mark D Chaffin, Zeina R Al Sayed, Matthew C Hill, Helene Mantineo, Elizabeth M Brydon, Zexian Zeng, X Shirley Liu, et al. Transfer learning enables predictions in network biology. *Nature*, 618(7965):616–624, 2023.

Haoran Wang, Xuanyi Zhang, Shuangsang Fang, Longke Ran, Ziqing Deng, Yong Zhang, Yuxiang Li, and Shaoshuai Li. Okr-cell: Open world knowledge aided single-cell foundation model with robust cross-modal cell-language pre-training. *bioRxiv*, pp. 2026–01, 2026.

Suyuan Zhao, Jiahuan Zhang, Yushuai Wu, Yizhen Luo, and Zaiqing Nie. Langcell: Language-cell pre-training for cell identity understanding. *arXiv preprint arXiv:2405.06708*, 2024.

Grace XY Zheng, Jessica M Terry, Phillip Belgrader, Paul Ryvkin, Zachary W Bent, Ryan Wilson, Solongo B Ziraldo, Tobias D Wheeler, Geoff P McDermott, Junjie Zhu, et al. Massively parallel digital transcriptional profiling of single cells. *Nature communications*, 8(1):14049, 2017.