COMPRESSING SINGLE-CELL FOUNDATION MODELS VIA SPARSE MULTI-LEVEL DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Single-cell foundation models provide reusable representations for biological analysis, but their size and inference cost limit deployment. We introduce **scPACK** (Parameter-efficient Architecture Compression with Knowledge transfer), a framework for compressing these models while transferring their learned capabilities. The framework represents gene embeddings as sparse combinations of shared dictionary atoms and reduces backbone width and depth. Training combines ground-truth supervision and teacher-prediction matching for the same self-supervised tasks, alongside gene, cell, and internal feature matching; Nash multi-task learning adjusts the objective weights. Evaluated on scGPT, the 3.743M-parameter student retains 89.15% of the teacher's fine-tuned Macro-F1 on the multi-study Broad collection and 99.49% on the Cardiac dataset, held blind during framework development, while using only 7.29% of the teacher's deployed parameters. Under matched inference conditions, its forward throughput is 2.86× the teacher's. Evaluation also covers few-shot annotation, clustering, donor mixing, and masked-expression reconstruction. Cell representations remain close to the teacher's even though nearest-neighbor relationships among gene embeddings change substantially. At the same training budget, additional teacher supervision improves annotation over ground-truth-only training, and dynamic weighting improves matching to the teacher's expression predictions during multi-objective training. With this framework, a compact model retains substantial teacher capability, supporting parameter-efficient single-cell deployment.

1 INTRODUCTION

Single-cell analysis distinguishes cell types and states across tissues, studies, and diseases. Single-cell foundation models (scFMs) learn reusable gene and cell representations from large transcriptomic collections, supporting several analyses rather than one fixed classifier (Cui et al., 2024; Hao et al., 2024; Wen et al., 2024). Beyond these immediate applications, reusable cellular representations are an important component of AI virtual-cell models, which aim to connect molecular measurements with cellular states and responses (Bunne et al., 2024). Making such representations practical to deploy broadens access to the knowledge learned during pretraining.

That value comes with a deployment cost. scFoundation has 100M parameters and CellFM has 800M (Hao et al., 2024; Zeng et al., 2025). For a genome-wide deletion screen processing 122.91 million original and simulated profiles, full-precision 316M-parameter Geneformer inference is estimated to take 32.8 days on twelve 80-GB A100 GPUs (Chen et al., 2026). This motivates a complementary question to scaling: how far can we reduce a pretrained model's deployed parameter count while preserving its downstream performance?

Knowledge distillation provides one route. In vision and language, students learn from teacher predictions and intermediate features; DistilBERT and TinyBERT show that shallower Transformers can retain strong task performance (Hinton et al., 2015; Romero et al., 2015; Sanh et al., 2019; Jiao et al., 2020). In biology, Implicit Structure Model (ISM) transfers structural information into a sequence-only protein model (Ouyang-Zhang et al., 2025). Geneformer quantization reduces numerical precision while retaining the architecture (Chen et al., 2026). MicroSC also studies compact scGPT students with factorized embeddings (Denvis, 2026). Task scores alone do not show whether compression also preserves the teacher's cell and gene representations.

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.

scGPT provides a strong starting point: scEval ranks it among its leading models when considering performance and usability (Liu et al., 2026). scGPT uses a dense embedding table with 60,697 entries—60,694 genes and three special tokens—each represented by a 512-dimensional vector. For each input gene, its embedding is added to an encoding of its expression value, and the resulting sequence is processed by the 12-layer, width-512 Transformer used in our teacher. The contextualized representation of the prepended CLS token serves as the cell representation. Two heads predict expression values at masked gene positions: gene expression prediction (GEP) uses contextual gene features, while gene expression prediction for cell modelling (GEPC) uses the cell representation and queries derived from gene embeddings (Cui et al., 2024).

Reducing Transformer depth, as in DistilBERT, leaves the embedding table unchanged. Here that table holds 31.08M values, or 60.5% of the teacher's parameters for cell embeddings and expression prediction. Substantial compression therefore requires addressing both the embedding table and the backbone.

We introduce **scPACK** to compress both components while retaining teacher capabilities. Shared gene-expression programs motivate representing each gene through a small combination of dictionary atoms (Kotliar et al., 2019). The resulting embeddings are combined with expression-value encodings and processed by a narrower, shallower backbone. Ground-truth supervision fits GEP/GEPC predictions to the data; teacher predictions and features guide transfer; Nash multi-task learning (Nash-MTL) adjusts the five objective weights (Navon et al., 2022). Using scGPT, we compare cell-level retention with changes in gene-embedding geometry, then measure inference cost and test the training choices. Our contributions are:

1. **A joint compression framework:** sparse gene embeddings, a smaller backbone, and dynamically weighted multi-level distillation.
2. **Multi-level transfer evidence:** two scGPT students assessed through cell and gene representations, downstream tasks, and inference costs.
3. **An insight into parameter efficiency:** our scGPT results point to substantial untapped potential for more parameter-efficient deployment of single-cell foundation models.

2 Related Work

Single-cell foundation models. Many scFMs combine input embeddings with a contextual backbone and self-supervised prediction, modeling genes within a cell or relationships between cells. scGPT learns gene expression prediction from over 33 million human cell profiles (Cui et al., 2024). scFoundation uses a 100M-parameter asymmetric Transformer and read-depth-aware reconstruction on over 50 million human profiles (Hao et al., 2024). CellPLM aggregates gene embeddings into cell tokens; its 82.4M-parameter model reconstructs masked expression from 11.4 million single-cell and spatial profiles, including tumor data (Wen et al., 2024). CellFM learns masked-expression recovery with an 800M-parameter ERetNet backbone on 100 million human cells (Zeng et al., 2025). Geneformer scales masked gene-identity prediction to 316M parameters and approximately 104 million human transcriptomes across tissues and diseases (Chen et al., 2026).

Compression and knowledge distillation. Pruning and quantization reduce stored weights or numerical precision (Han et al., 2016); distillation transfers predictions or features into a student (Hinton et al., 2015; Romero et al., 2015). DistilBERT reduces BERT's size by 40% while retaining 97% of its GLUE performance (Sanh et al., 2019). Four-layer TinyBERT retains over 96.8% of BERT-Base's GLUE performance with a model 7.5 times smaller and inference 9.4 times faster (Jiao et al., 2020). In proteins, ISM transfers structure-derived tokens to a sequence-only model, increasing long-range contact precision from 0.35 to 0.49 (Ouyang-Zhang et al., 2025). Geneformer quantization reduces the estimated deletion-screen time above from 32.8 to 5.9 days at the same batch size (Chen et al., 2026). MicroSC also describes factorized scGPT embeddings and distillation (Denvis, 2026). The cited version (v1) uses expression-bin logits for distillation but does not specify how they are obtained from scGPT's real-valued expression outputs. A numerical comparison would therefore require an assumed construction of these targets, so we discuss its design without reproducing its results.

Sparse representations. Dictionary learning represents vectors using a shared set of atoms with sparse coefficients (Mairal et al., 2010). Compositional code learning uses related sharing to compress

word embeddings (Shu & Nakayama, 2018). Both reduce embedding storage while preserving vocabulary entries. Gene-expression programs provide a biological motivation for shared components (Kotliar et al., 2019).

Multi-objective optimization. Nash-MTL addresses conflicting objectives through a bargaining formulation that chooses positive weights from shared-parameter gradients (Navon et al., 2022). The weights adapt to changing gradient scales and directions.

3 THE SCPACK FRAMEWORK

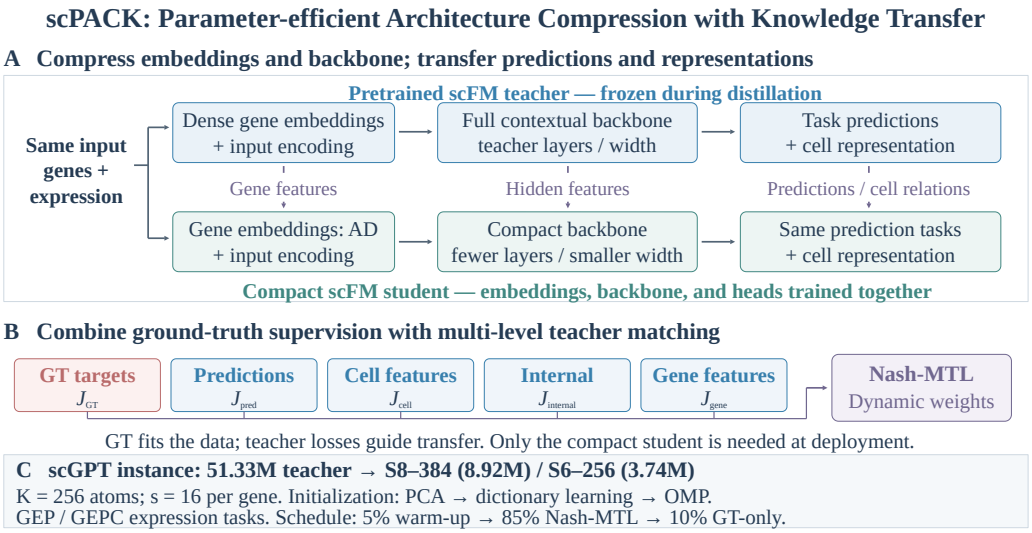


Figure 1: **scPACK transfers a frozen teacher into a compact student.** (A) Solid arrows show forward computation; dashed arrows mark feature and prediction matching. (B) Ground-truth (GT) targets and four teacher objectives train the student with Nash-MTL weights during the main distillation stage. (C) Architecture values and the schedule refer to scGPT. Here $J_{\text{pred}} = J_{\text{expr}}$ matches student and teacher outputs for gene expression prediction (GEP) and gene expression prediction for cell modelling (GEPC). Principal component analysis (PCA) and orthogonal matching pursuit (OMP) are initialization-only. The teacher count covers shared cell/GEP/GEPC functions: 51.33M, or 51.86M including an unused classification head.

3.1 FRAMEWORK DESIGN

Figure 1 summarizes scPACK. Gene embeddings formed from a sparse latent dictionary (SLD) are combined with expression-value encodings and passed to a smaller backbone. Teacher and student receive the same input; ground-truth targets and teacher predictions and features jointly supervise the student, with dynamic weights. The design requires an explicit gene-token table, a compressible backbone, and corresponding genes and prediction tasks. Feature adapters accommodate different widths.

Compress embeddings and backbone together. Let G be the number of ordinary genes, K the dictionary size, d the student width, and s the maximum number of atoms per gene. For positive integers with $1 \le s \le K$, the pre-normalization ordinary-gene embedding matrix is

$$\widetilde{E}_S = AD, \quad A \in \mathbb{R}^{G\times K}, \quad D \in \mathbb{R}^{K\times d}, \quad \|A_{g,:}\|_0 \le s \quad (g = 1, \dots, G). \tag{1}$$

The row $A_{g,:}$ gives gene g's coefficients, $D_{k,:}$ is atom k, and $\|\cdot\|_0$ counts nonzero entries. For each gene, store $0 \le s_g \le s$ distinct indices $I_{gj} \in \{1, \dots, K\}$ and coefficients $c_{gj} = A_{g,I_{gj}}$. With layer normalization (LN), the embedding is

$$e_S(g) = \mathrm{LN}(A_{g,:}D) = \mathrm{LN}\left(\sum_{j=1}^{s_g} c_{gj} D_{I_{gj},:}\right). \tag{2}$$

The atom indices remain fixed after initialization; the selected coefficients, dictionary atoms, and LayerNorm parameters are trainable. Special tokens use a separate lookup. Although A is written as a $G \times K$ matrix, sparse storage keeps at most s coefficients per gene, requiring $Gs + Kd$ floating-point values rather than Gd for a dense embedding matrix, plus storage for the atom indices. The resulting gene embeddings are combined with expression-value encodings and passed to a narrower, shallower backbone.

Transfer tasks and representations jointly. Ground-truth (GT) supervision fits self-supervised task predictions to data-derived targets. Prediction-level knowledge distillation (KD) matches them to teacher predictions on the same inputs. Matching predictions alone leaves the underlying representations weakly constrained, so gene, cell, and internal losses also align the features from which those predictions are formed. Internal matching includes selected hidden layers and cell–cell relations. Separate training-only adapters map student features to teacher width; the compact model needs none of them at inference.

Update the objective weights. These objectives need not favor the same update. Let J_i, $i = 1, \ldots, 5$, denote the GT, prediction, cell, internal, and gene objectives. For shared representation parameters θ_{sh}, form $\boldsymbol{g}_i = \nabla_{\theta_{\text{sh}}} J_i$ and $Q_{ij} = \boldsymbol{g}_i^\top \boldsymbol{g}_j$. Nash-MTL seeks positive weights (Navon et al., 2022):

$$Q\alpha = \alpha^{-1}, \qquad Q \in \mathbb{R}^{5\times 5}, \quad \alpha \in \mathbb{R}^5, \quad \alpha_i > 0. \tag{3}$$

The reciprocal is elementwise. Between weight updates, the student minimizes $\sum_i \alpha_i J_i$. The frozen teacher supplies targets but receives no gradients. The objective weights affect training only; deployment retains just the compact student.

3.2 Implementation with scGPT

For scGPT, $G = 60{,}694$ ordinary genes and three special tokens give $V = 60{,}697$. We construct S8–384 with eight layers of width 384 and S6–256 with six layers of width 256; both use $K = 256$ and $s = 16$, eight attention heads, and feed-forward width equal to hidden width.

Initialization. We subtract the mean of all V teacher embedding rows, then use principal component analysis (PCA) to project to student width. No standard-deviation or vector-norm scaling is applied. Dictionary learning fits the projected ordinary-gene vectors; orthogonal matching pursuit (OMP) selects at most s atoms and fits their coefficients per gene. Special tokens bypass dictionary fitting. PCA and OMP are initialization-only; the student Transformer starts at random (Appendix A.1).

Expression and feature targets. For masked gene positions $\mathcal{M}$ (excluding CLS and padding), with $|\mathcal{M}| > 0$, define mean squared error (MSE):

$$\ell_{\mathcal{M}}(p, y) = \frac{1}{|\mathcal{M}|} \sum_{u \in \mathcal{M}} (p_u - y_u)^2. \tag{4}$$

For task $h \in \{\text{GEP}, \text{GEPC}\}$, student and teacher predictions p_S^h, p_T^h use the same masked input. Each student prediction is compared separately with the ground-truth binned expression value y and the corresponding teacher prediction:

$$\ell_{\text{GT-}h} = \ell_{\mathcal{M}}(p_S^h, y), \qquad \ell_{\text{KD-}h} = \ell_{\mathcal{M}}(p_S^h, p_T^h). \tag{5}$$

An empty mask gives zero loss. Here u indexes a cell–token position and y contains the expression-bin values before masking. Predictions are real-valued, not bin-class logits; prediction KD compares outputs, not reconstruction losses. Cell and gene matching use cosine distance after separate bias-free adapters to width 512. Hidden matching uses normalized-feature MSE at five layer pairs. Relation matching compares similarity matrices for the same groups of eight cells, with the actual cell count used for an incomplete group. Appendix A.2 specifies all reductions.

The eight components form five objectives, with fixed internal coefficients selected in early weighted-sum pilot runs:

$$\begin{aligned} J_{\text{GT}} &= \ell_{\text{GT-GEP}} + \ell_{\text{GT-GEPC}}, & J_{\text{expr}} &= \ell_{\text{KD-GEP}} + 0.5\ell_{\text{KD-GEPC}}, \\ J_{\text{cell}} &= \ell_{\text{cell}}, & J_{\text{internal}} &= 0.5\ell_{\text{hidden}} + 0.25\ell_{\text{relation}}, \\ J_{\text{gene}} &= \ell_{\text{gene}}. \end{aligned} \tag{6}$$

These coefficients stay fixed after grouping; Nash-MTL adjusts only the weights between objectives. Every 20 updates, we form Q from gene/value-encoder and Transformer gradients and solve for the five weights with damped Newton. Heads and adapters receive weighted-loss gradients but do not enter Q.

Training stages. Our goal is a compact model that retains downstream capability and fits expression data. Teacher predictions and features guide transfer; the final stage uses only the GEP/GEPC ground-truth losses. Warm-up, dynamic distillation, and GT-only consolidation use approximately 5%, 85%, and 10% of processed records, respectively (Appendix A.3).

4 Experimental Setup

4.1 Experimental Configuration

Data and models. All distillation and evaluation data are from CELLxGENE Discover. Main distillation uses 6,580,200 profiles from 450 dataset releases; ID contains 19,800 record-held-out profiles for preliminary calibration. Broad contains 120,000 profiles from 172 releases spanning multiple tissues and studies. Cardiac contains 62,705 disease-associated myocardial nuclei from the CELLxGENE release of Maatz et al. (2025) and was held blind during compression choices. Neither Broad nor Cardiac was used to select the compression architecture or distillation hyperparameters. Appendix B.1 details the scGPT-based temporal cutoff and source selection.

The support/validation/query splits contain 77,018/22,419/20,563 Broad profiles and 36,406/12,578/13,721 Cardiac profiles, with dataset-scoped donors disjoint across splits. Broad query has 410 labels, including 20 absent from support; Cardiac has nine. Support fits readouts, validation selects the FT checkpoint, and query provides final scores. Baselines are the scGPT teacher and a support-fitted 50-component PCA expression representation for kNN, linear probes, and clustering, following the emphasis on simple baselines in scFM evaluation (Kedzierska et al., 2025).

Training and compute. Each main student is trained for 25,704 updates with effective batch size 256. Preliminary calibration selected architecture-specific learning rates of 3×10^{-4} for S8–384 and 5×10^{-4} for S6–256 using final-step linear centered kernel alignment (CKA) between teacher and student cell representations on ID (Kornblith et al., 2019); these rates were then fixed for full-budget distillation. AdamW (Loshchilov & Hutter, 2019) uses weight decay 0.01, gradient clipping at norm one, 1,285 learning-rate warm-up updates, and cosine decay. Training uses bfloat16 and FlashAttention-2 (Dao, 2024) on one RTX A6000. Inputs contain at most 1,199 genes plus CLS. Nonzero raw counts are quantile-binned within each cell to levels 1–50 under scGPT's $n_{\text{bins}} = 51$ convention; zero-count genes are omitted, and binning precedes gene sampling (Appendix B.2).

Table 1: **Model size and storage.** Counts use the same cell/GEP/GEPC functions as Figure 1. Files include sparse indices; training adapters are excluded from deployment.

Model	Layers/width	Params (M)	Teacher ratio	File (MiB)
Teacher	12/512	51.329	100.00%	195.87
S8–384	8/384	8.921	17.38%	35.92
S6–256	6/256	3.743	7.29%	16.16

4.2 Evaluation Procedures

Representation and tasks. Linear CKA compares teacher and student representations for the same cells after column centering. Cell CKA uses the base models before task-specific adaptation on each full evaluation set; local cell analysis uses shared 15,000-cell ID/Broad subsets. Neighbor Recall@k averages intersection size divided by k, and Jaccard@k averages intersection divided by union, for $k = 10, 20, 50$. Gene analysis uses all 60,694 ordinary genes after gene-encoder LayerNorm, before adding expression-value encodings or entering the Transformer, without training adapters.

For cell-type annotation, cosine k-nearest neighbors (kNN, $k = 15$) and linear probing (LP) use frozen representations; end-to-end fine-tuning (FT) runs for five epochs. LP uses support-fitted

standardization and balanced class weights; FT uses unweighted cross-entropy and selects checkpoints by validation Macro-F1. Macro-F1 is primary; accuracy is also reported. Few-shot kNN uses six support fractions and three support-set draws shared across models. Clustering and donor mixing use a shared 5,000-query-profile subset per dataset, with normalized mutual information (NMI), adjusted Rand index (ARI), cell-type average silhouette width (ASW-cell), and within-cell-type donor mixing (ASW-donor) (Luecken et al., 2022). Masked-expression reconstruction reports GEP/GEPC MSE against ground truth, pooled over masked positions in each full evaluation set. Appendix B defines these evaluations; Appendix C.2 reports donor-composition sensitivity.

Inference. Inference is benchmarked in isolated RTX A6000 processes using identical 1,200-token tensors, bfloat16, the same cell/GEP/GEPC forward functions, and batch sizes 1, 8, 32, and 64. After 50 warm-up passes, 200 CUDA-synchronized forward passes measure throughput and latency; memory is peak PyTorch allocation. Measurements are forward-only and exclude data loading and host-to-device transfer.

4.3 Ablation Design

At a shared 1,645,050-record budget (6,426 updates), W versus G tests teacher supervision: G is GT-only, whereas W adds teacher supervision with weights calibrated once after warm-up. N versus W tests weight adaptation: N refreshes Nash weights, while W keeps that initial calibration. All use the S6–256 architecture, teacher-derived sparse initialization, input order, and one training seed; N/W also share post-warm-up model, optimizer, adapter, and random states. We compare their unweighted losses over 5,462 matched multi-objective updates (323–5,784), then evaluate both models immediately before and after the final GT-only stage on the same 1,024 ID profiles with identical masks (Appendix D).

5 Cell Representations and Downstream Performance

5.1 Cell-Level Similarity

We first compare teacher and student cell representations before task-specific fine-tuning. Under linear CKA, S8–384/S6–256 score 0.8636/0.8856 on ID, 0.8747/0.8785 on Broad, and 0.9861/0.9851 on Cardiac (Figure 2a). Nearest-neighbor overlap provides a complementary local comparison: S8–384 has higher Jaccard at all three neighborhood sizes on ID and Broad, including 0.4207 versus 0.4004 at $k = 10$ on Broad (Figure 2b). Both students show strong global representation similarity to the teacher, while S8–384 preserves somewhat more of the teacher's local cell neighbors. We next test whether these similarities are reflected in downstream performance.

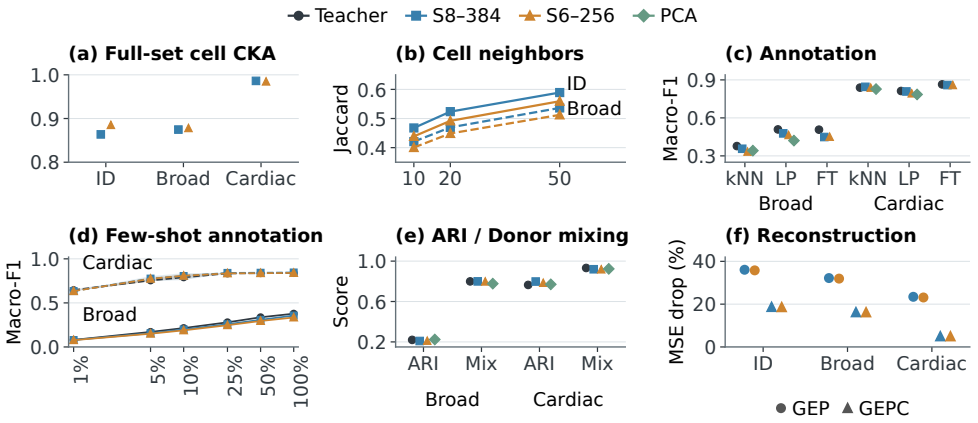


Figure 2: **Cell representations and task performance.** (a) Cell CKA on full sets; (b) local overlap on shared 15,000-cell ID/Broad subsets. (c) kNN, linear-probe, and fine-tuned Macro-F1; Table 2 also reports accuracy. (d) Few-shot means and min–max ranges over three support draws. (e) Clustering and donor mixing on 5,000 query profiles per dataset. (f) GT reconstruction MSE reduction relative to the teacher.

5.2 Annotation and Label Efficiency

Table 2: **Cell-type annotation.** Each entry is **Macro-F1 / accuracy** on the full query set; higher is better. k-nearest neighbors (kNN) and linear probe (LP) use frozen representations; fine-tuning (FT) updates the backbone. PCA has no FT result.

Dataset	Model	kNN	Linear probe	Fine-tuning
Broad	Teacher	0.3774 / 0.6977	0.5089 / 0.7261	0.5069 / 0.7868
Broad	S8–384	0.3562 / 0.6769	0.4777 / 0.7033	0.4489 / 0.7439
Broad	S6–256	0.3344 / 0.6672	0.4654 / 0.6905	0.4520 / 0.7340
Broad	PCA	0.3414 / 0.6967	0.4206 / 0.6569	–
Cardiac	Teacher	0.8389 / 0.9815	0.8123 / 0.9754	0.8644 / 0.9871
Cardiac	S8–384	0.8452 / 0.9789	0.8086 / 0.9730	0.8591 / 0.9834
Cardiac	S6–256	0.8396 / 0.9772	0.7944 / 0.9683	0.8600 / 0.9865
Cardiac	PCA	0.8276 / 0.9795	0.7851 / 0.9582	–

While retaining 89.15% of the teacher's FT Macro-F1 on Broad and 99.49% on blind Cardiac, S6–256 uses only 7.29% of the teacher's deployed parameters. Absolute F1 is 0.4520 versus 0.5069 on Broad and 0.8600 versus 0.8644 on Cardiac (Table 2). The drops are 5.50 and 0.44 percentage points, respectively. Frozen LP retains 91.45% and 97.79% of the corresponding teacher scores. S8–384 improves Broad frozen-readout scores over S6–256, but not FT scores.

On Broad, FT increases accuracy for all neural models while Macro-F1 stays similar or falls relative to LP. This is consistent with different class weighting: LP balances classes, whereas FT weights examples equally. Appendix B.2 discusses this accuracy–Macro-F1 difference alongside the fitting procedures. PCA slightly exceeds S6–256 on Broad kNN, but both students exceed PCA on Broad LP and both Cardiac frozen readouts.

Few-shot results show close Cardiac performance across label budgets (Figure 2d). The normalized F1 area over log support fraction is 0.7746/0.7828/0.7802 for Teacher/S8–384/S6–256, with overlapping three-draw ranges. Broad values are 0.2220/0.2057/0.1965; the teacher remains ahead.

5.3 Clustering and Expression Reconstruction

Cardiac ARI is 0.7977/0.7863 for S8–384/S6–256, versus 0.7643 for the teacher. Students also have higher ASW-cell, but donor mixing is slightly lower, about 0.919 versus 0.932. Broad ARI is 0.2089/0.2129 versus 0.2199. Cell-type separation and donor mixing thus show different changes after compression. Cardiac UMAPs provide qualitative context (Appendix F.2).

Both students have lower masked-expression reconstruction error than the teacher on all three sets. S6–256 reduces GEP MSE by 35.79%, 31.94%, and 23.12% on ID, Broad, and Cardiac; GEPC reductions are 18.55%, 16.13%, and 4.91%. These are errors against expression-bin values, distinct from teacher-matching losses. Full reconstruction values are in Appendix C.

6 Gene Representations under Compression

The cell-level results show transfer at the backbone output. We also ask whether gene embeddings before adding expression-value encodings and entering the Transformer preserve the teacher's gene-to-gene relationships. Figure 3 shows substantial changes despite positive global similarity. Across 60,694 genes, CKA is 0.7250 for S8–384 and 0.7211 for S6–256, while neighbor Recall@10 is 0.0628 and 0.0764. The smaller student preserves more nearest gene neighbors despite its slightly lower global CKA. This differs from the cell-neighbor ranking in Section 5.

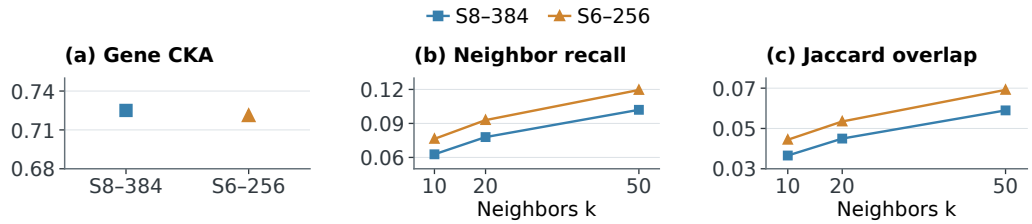


Figure 3: **Gene representation similarity.** Global CKA and local neighbor recall/Jaccard measure different properties of the same gene embeddings. CKA measures global similarity; recall and Jaccard measure the overlap of nearest-neighbor sets.

Together with Section 5, these results show that strong cell-level retention can coexist with substantial changes in nearest-neighbor relationships among gene embeddings. Thus, preserving the evaluated cell-level capabilities does not require the student to reproduce the teacher's local gene-embedding geometry exactly. Because the students combine width reduction with sparse dictionary parameterization, these experiments do not isolate the contribution of either change to the altered gene-embedding geometry.

7 Model Size and Inference Efficiency

We next ask whether the smaller parameter budget yields a practical inference benefit under the matched conditions in Section 4.

Table 3: **A6000 inference at batch 64.** Identical 1,200-token inputs and cell/GEP/GEPC functions. Latency is per batch; memory is peak PyTorch allocation.

Model	Cells/s	Speedup	Latency (ms)	Memory (GiB)
Teacher	444.30	1.00×	144.05	1.381
S8–384	612.13	1.38×	104.55	3.718
S6–256	1271.75	2.86×	50.32	2.491

S6–256 reduces model-file size from 195.87 to 16.16 MiB. At batch 64, it processes 1,271.75 cells/s versus 444.30 for the teacher, giving 2.86× forward throughput. S8–384 reaches 612.13 cells/s. Both students are faster at every tested batch size (Figure 4).

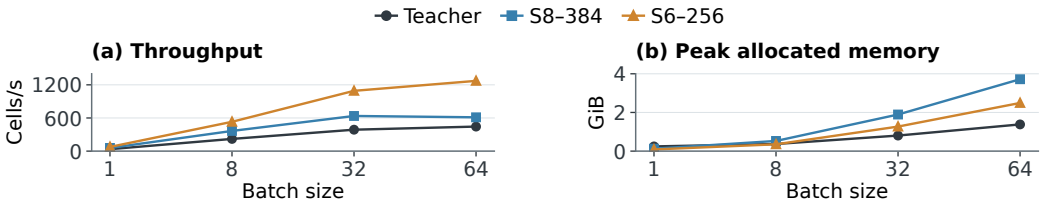


Figure 4: **Inference efficiency under matched conditions.** Throughput and peak PyTorch allocation on an RTX A6000, with the same inputs, precision, and cell/GEP/GEPC functions.

Peak allocation for S6–256 falls from the teacher's 0.243 to 0.086 GiB at batch one, but rises from 1.381 to 2.491 GiB at batch 64. Its sparse lookup creates a temporary $[B, T, s, d]$ tensor, where B is batch size and T is padded token length. This temporary tensor grows with batch size and helps explain the observed memory pattern. Full timing and memory values are in Appendix E.

8 Ablation Studies

Matched-budget ablations ask two questions: whether teacher supervision improves a compact model beyond ground-truth-only training, and whether dynamic objective weights improve matching to the teacher's predictions when the objectives conflict. We address them separately using the comparisons in Section 4.3.

8.1 What Does Teacher Supervision Add?

Fixed-weight W improves both kNN and LP Macro-F1 over GT-only G on Broad and Cardiac (Figure 5a). Broad LP Macro-F1 rises from 0.3253 to 0.3851 and kNN from 0.1707 to 0.2246; Cardiac LP rises from 0.7614 to 0.7727 and kNN from 0.7122 to 0.8180. Thus, fixed-weight teacher supervision improves annotation at the same data and update budget, without the additional effect of dynamic weight updates. Appendix D.1 reports every variant and task.

8.2 Dynamic Weighting under Competing Objectives

Across the multi-objective stage, N lowers mean unweighted loss J_{expr} for matching the teacher's GEP/GEPC predictions from 44.63 to 25.13 (43.7%) relative to W, while mean J_{GT} rises from 329.22 to 357.87 (Figure 5b,c). Dynamic weighting therefore shifts the trade-off toward prediction matching rather than improving every objective. Appendix D.1 reports the final downstream results.

The gradient analysis helps explain this trade-off. At 264 of N's 274 recorded gradient evaluations (96.35%), J_{GT} and J_{expr} gradients have negative cosine similarity, with median -0.9427. On these same N states, dynamic weights give $(Q\alpha)_i > 0$ for all five objectives in 274/274 evaluations, versus 36/274 for W's initial fixed weights (Figure 5d). This is a local first-order diagnostic at matched states, not a comparison of the actual optimizer trajectories (Appendix D.3).

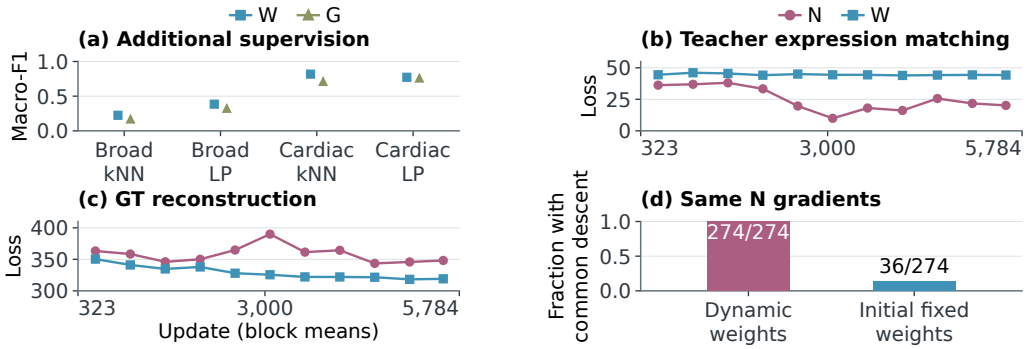


Figure 5: **Teacher supervision and dynamic weighting.** (a) Annotation Macro-F1 for fixed-weight distillation (W) and GT-only training (G). (b,c) Unweighted J_{expr} and J_{GT} for N/W, averaged over fixed 500-update blocks (462 updates in the final block). (d) At each recorded N gradient state, the same gradients receive the current dynamic weights or W's initial fixed weights. The fraction with $(Q\alpha)_i > 0$ for all five objectives indicates a common local first-order descent direction.

Fixed-input checkpoints show how the final GT-only stage changes this balance (Appendix D.2, Table 13). On the same 1,024 ID profiles and masks, J_{GT} decreases from 340.78/310.38 before consolidation to 293.79/292.74 afterward for N/W, whereas J_{expr} increases from 13.72/36.74 to 97.53/98.10. Thus, on this fixed diagnostic set, the final stage improves reconstruction against ground truth while weakening the match to the teacher's predictions. Testing partial teacher supervision in the final stage could assess whether more of the earlier matching can be retained.

9 Conclusion

scPACK provides a joint embedding–backbone compression framework for single-cell foundation models. In scGPT, S6–256 retains 89.15% and 99.49% of the teacher's fine-tuned Macro-F1 on Broad and Cardiac, respectively, while using only 7.29% of the teacher's deployed parameters and achieving 2.86× the teacher's forward throughput under matched inference conditions. Cell representations remain highly similar to the teacher even though nearest-neighbor relationships among gene embeddings change substantially, showing that strong cell-level retention can coexist with substantial gene-level reorganization. In matched-budget ablations, teacher supervision improves annotation over ground-truth-only training, while dynamic weighting improves matching to the teacher's expression predictions during the multi-objective stage. More broadly, our scGPT results point to substantial untapped potential for parameter-efficient scFM deployment.

AI USE STATEMENT

AI tools, including ChatGPT and Codex, assisted with research discussion, literature search and summarization, code development and review, result checks, manuscript polishing and editing, and figure preparation. Experimental values come from model runs and evaluations, not synthetic results generated for the manuscript.

ETHICS STATEMENT

This study reuses human transcriptomic datasets and pretrained models. It involves no new participant recruitment or sample collection. The models are evaluated for computational single-cell analysis, not clinical decision-making.

REPRODUCIBILITY STATEMENT

Section 3 defines the framework and its scGPT implementation. Section 4 and Appendices A–B specify initialization, training settings, data selection, and evaluation procedures. Appendices C–F provide additional results, ablation analyses, resource accounting, and qualitative examples. Source code and trained weights will be made publicly available in the future but have not yet been packaged or uploaded at the time of submission.

REFERENCES

Charlotte Bunne, Yusuf Roohani, Yanay Rosen, Ankit Gupta, Xikun Zhang, Marcel Roed, Theo Alexandrov, Mohammed AlQuraishi, Patricia Brennan, Daniel B. Burkhardt, Andrea Califano, Jonah Cool, Abby F. Dernburg, Kirsty Ewing, Emily B. Fox, Matthias Haury, Amy E. Herr, Eric Horvitz, Patrick D. Hsu, Viren Jain, Gregory R. Johnson, Thomas Kalil, David R. Kelley, Shana O. Kelley, Anna Kreshuk, Tim Mitchison, Stephani Otte, Jay Shendure, Nicholas J. Sofroniew, Fabian Theis, Christina V. Theodoris, Srigokul Upadhyayula, Marc Valer, Bo Wang, Eric Xing, Serena Yeung-Levy, Marinka Zitnik, Theofanis Karaletsos, Aviv Regev, Emma Lundberg, Jure Leskovec, and Stephen R. Quake. How to build the virtual cell with artificial intelligence: Priorities and opportunities. *Cell*, 187(25):7045–7063, 2024. doi: 10.1016/j.cell.2024.11.015.

Han Chen, Madhavan S. Venkatesh, Javier Gómez Ortega, Siddharth V. Mahesh, Tarak N. Nandi, Ravi K. Madduri, Karin Pelka, and Christina V. Theodoris. Scaling and quantization of large-scale foundation model enables resource-efficient predictions in network biology. *Nature Computational Science*, 6:450–463, 2026. doi: 10.1038/s43588-026-00972-4.

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. doi: 10.1038/s41592-024-02201-0.

Tri Dao. FlashAttention-2: Faster attention with better parallelism and work partitioning. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/98ed250b203d1ac6b24bbcf263e3d4a7-Abstract-Conference.html.

Olivia Denvis. MicroSC: A tiny, performant single-cell foundation model. Research Square preprint, 2026. URL https://www.researchsquare.com/article/rs-10410549/v1. Version 1, July 21, 2026.

Song Han, Huizi Mao, and William J. Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. In *International Conference on Learning Representations*, 2016. URL https://arxiv.org/abs/1510.00149v5.

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. doi: 10.1038/s41592-024-02305-7.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. arXiv:1503.02531, 2015. URL https://arxiv.org/abs/1503.02531v1.

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. TinyBERT: Distilling BERT for natural language understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 4163–4174, 2020. doi: 10.18653/v1/2020.findings-emnlp.372. URL https://aclanthology.org/2020.findings-emnlp.372/.

Kasia Z. Kedzierska, Lorin Crawford, Ava P. Amini, and Alex X. Lu. Zero-shot evaluation reveals limitations of single-cell foundation models. *Genome Biology*, 26:101, 2025. doi: 10.1186/s13059-025-03574-x.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 3519–3529, 2019. URL https://proceedings.mlr.press/v97/kornblith19a.html.

Dylan Kotliar, Adrian Veres, M. Aurel Nagy, Shervin Tabrizi, Eran Hodis, Douglas A. Melton, and Pardis C. Sabeti. Identifying gene expression programs of cell-type identity and cellular activity with single-cell RNA-Seq. *eLife*, 8:e43803, 2019. doi: 10.7554/eLife.43803.

Tianyu Liu, Kexing Li, Yuge Wang, Hongyu Li, and Hongyu Zhao. Evaluating the utilities of foundation models in single-cell data analysis. *Advanced Science*, 13(27):e14490, 2026. doi: 10.1002/advs.202514490.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://arxiv.org/abs/1711.05101v3.

Malte D. Luecken, M. Büttner, K. Chaichoompu, A. Danese, M. Interlandi, M. F. Mueller, D. C. Strobl, L. Zappia, M. Dugas, M. Colomé-Tatché, and Fabian J. Theis. Benchmarking atlas-level data integration in single-cell genomics. *Nature Methods*, 19(1):41–50, 2022. doi: 10.1038/s41592-021-01336-8.

Henrike Maatz, Eric L. Lindberg, Eleonora Adami, et al. The cellular and molecular cardiac tissue responses in human inflammatory cardiomyopathies after SARS-CoV-2 infection and COVID-19 vaccination. *Nature Cardiovascular Research*, 4(3):330–345, 2025. doi: 10.1038/s44161-025-00612-6.

Julien Mairal, Francis Bach, Jean Ponce, and Guillermo Sapiro. Online learning for matrix factorization and sparse coding. *Journal of Machine Learning Research*, 11(2):19–60, 2010. URL https://jmlr.org/papers/v11/mairal10a.html.

Leland McInnes, John Healy, Nathaniel Saul, and Lukas Großberger. UMAP: Uniform manifold approximation and projection. *Journal of Open Source Software*, 3(29):861, 2018. doi: 10.21105/joss.00861.

Aviv Navon, Aviv Shamsian, Idan Achituve, Haggai Maron, Kenji Kawaguchi, Gal Chechik, and Ethan Fetaya. Multi-task learning as a bargaining game. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 16428–16446, 2022. URL https://proceedings.mlr.press/v162/navon22a.html.

Jeffrey Ouyang-Zhang, Chengyue Gong, Yue Zhao, Philipp Krähenbühl, Adam R. Klivans, and Daniel J. Diaz. Distilling structural representations into protein sequence models. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/ea0b5818ae9255ee1fb1e3b4442d2ffe-Abstract-Conference.html.

Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. FitNets: Hints for thin deep nets. In *International Conference on Learning Representations*, 2015. URL https://arxiv.org/abs/1412.6550v4.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. arXiv:1910.01108, 2019. URL https://arxiv.org/abs/1910.01108v4. Version 4, revised March 2020.

Raphael Shu and Hideki Nakayama. Compressing word embeddings via deep compositional code learning. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=BJRZzFlRb.

Hongzhi Wen, Wenzhuo Tang, Xinnan Dai, Jiayuan Ding, Wei Jin, Yuying Xie, and Jiliang Tang. CellPLM: Pre-training of cell language model beyond single cells. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/173bcdbd820edbe43284e0941109ada9-Abstract-Conference.html.

Yuansong Zeng, Jiancong Xie, Ningyuan Shangguan, Zhuoyi Wei, Wenbing Li, Yun Su, Shuangyu Yang, Chengyang Zhang, Jinbo Zhang, Nan Fang, Hongyu Zhang, Yutong Lu, Huiying Zhao, Jue Fan, Weijiang Yu, and Yuedong Yang. CellFM: a large-scale foundation model pre-trained on transcriptomics of 100 million human cells. *Nature Communications*, 16:4679, 2025. doi: 10.1038/s41467-025-59926-5.

A Method and Training Details

A.1 Architecture and Initialization

All models use eight attention heads, ReLU feed-forward blocks with the same width as their hidden features, post-norm, dropout 0.2, a continuous-value encoder, and CLS pooling. Table 1 gives their layer counts and widths; Table 4 gives the remaining main-student settings. The vocabulary contains 60,694 ordinary genes and three special tokens. Student Transformer weights start at random.

Table 4: Main-student training settings. Learning rates were selected by preliminary ID calibration; both runs use one RTX A6000.

Setting	S8–384	S6–256
Learning rate	3×10^{-4}	5×10^{-4}
Weight decay	0.01	0.01
Optimizer betas	(0.9, 0.999)	(0.9, 0.999)
Effective batch	256	256
Microbatch / accumulation	128 / 2	256 / 1
Updates	25,704	25,704
Records processed	6,580,200	6,580,200
LR warm-up updates	1,285	1,285
Maximum tokens	1,200	1,200
Training seed	20260824	20260824
Nash refresh interval	20 updates	20 updates

For the raw teacher gene embedding table $E_T \in \mathbb{R}^{V\times 512}$, initialization computes

$$\mu = \frac{1}{V}\sum_{v=1}^{V} E_{T,v,:}, \qquad E_c = E_T - \mathbf{1}_V \mu, \qquad Y = E_c U_d, \tag{7}$$

where $\mathbf{1}_V$ is a column of ones and $U_d \in \mathbb{R}^{512\times d}$ contains the eigenvectors of $E_c^\top E_c/(V-1)$ with the d largest eigenvalues. Thus $Y \in \mathbb{R}^{V\times d}$. Both student widths satisfy $d < 512$. The mean includes special tokens; their projected rows then initialize a separate lookup and are excluded from dictionary fitting.

Mini-batch dictionary learning fits the ordinary-gene rows of Y, using coordinate descent, 256 atoms, regularization $\lambda_{\text{dict}} = 1$, a limit of 500 passes, batch size 1,024, and seed 20260824. OMP encodes genes in chunks of 2,048, selects at most 16 atoms per gene, and refits coefficients over the selected atoms. Selected indices stay fixed during training; coefficients, atoms, and gene-encoder LayerNorm parameters are trainable. Unused slots have index -1 and zero contribution. Equations use one-based indices; Appendix F.1 uses zero-based atom IDs.

A.2 Feature Losses and Their Reductions

The GEP/GEPC reconstruction and prediction-matching losses are defined in Equations 4–6. Here we specify the four feature losses. Let B be the batch size, $z_b^S \in \mathbb{R}^d$, $z_b^T \in \mathbb{R}^{512}$ the CLS vectors for cell b, and $e^S(v), e^T(v)$ the gene-encoder vectors for token v, all written as columns. For a nonempty selected token set $\mathcal{U}$,

$$\ell_{\text{cell}} = \frac{1}{B}\sum_{b=1}^{B}[1 - \cos(P_c z_b^S, z_b^T)], \tag{8}$$

$$\ell_{\text{gene}} = \frac{1}{|\mathcal{U}|}\sum_{v\in\mathcal{U}}[1 - \cos(P_g e^S(v), e^T(v))]. \tag{9}$$

The bias-free adapters $P_c, P_g \in \mathbb{R}^{512\times d}$ map student features to teacher width. We use unique non-padding token IDs in each batch, capped at 4,096; above the cap, we sample using the training seed plus update step and sort the selected IDs. Special tokens can be included. An empty set gives zero gene loss. Cosines use numerical stabilization.

For fixed student/teacher layer pairs $\mathcal{P}$ and non-padding positions $\mathcal{T}_b$, including CLS, hidden matching uses a third adapter $P_h \in \mathbb{R}^{512\times d}$:

$$\ell_{\text{hidden}} = \frac{1}{|\mathcal{P}|} \sum_{(\ell_S,\ell_T)\in\mathcal{P}} \frac{\sum_{b=1}^{B}\sum_{u\in\mathcal{T}_b} \| \operatorname{LN}(P_h H^S_{b,u,\ell_S}) - \operatorname{LN}(H^T_{b,u,\ell_T})\|_2^2}{512\sum_{b=1}^{B}|\mathcal{T}_b|}. \tag{10}$$

Here $H^S \in \mathbb{R}^d, H^T \in \mathbb{R}^{512}$ denote position-level vectors; this LayerNorm has no learned scale or offset. The five zero-based layer pairs are $(0,0),(2,4),(4,6),(5,8),(7,11)$ for S8–384 and $(0,1),(1,3),(2,5),(4,9),(5,11)$ for S6–256.

For r consecutive cells, let $Z_S \in \mathbb{R}^{r\times d}$ and $Z_T \in \mathbb{R}^{r\times 512}$ contain row-normalized CLS vectors. Relation matching needs no adapter:

$$\ell^{(r)}_{\text{relation}} = \frac{1}{r^2}\|Z_S Z_S^\top - Z_T Z_T^\top\|_F^2, \qquad 1 \le r \le 8. \tag{11}$$

The squared Frobenius norm sums over all entries, including self-pairs. Groups normally have eight cells; a final incomplete group uses its actual size. Grouping does not use cell-type labels.

Batch reductions. Within a microbatch, expression and hidden losses average over masked and non-padding positions, respectively; cell and gene losses average over cells and selected token IDs. Main-student training averages relation losses over groups in each microbatch and weights accumulated microbatch losses by cell count. N/W/G training instead pools components by the full update's masked-position, token, cell, or relation-pair counts. Gene losses are averaged over each selected set, then weighted by cell count. N and W use identical reductions. Training-curve means average update-level objectives; checkpoint evaluation pools component errors and counts.

A.3 Training Stages and Weight Updates

Main training uses one pass through the records. Architecture-specific learning rates were selected by final-step ID cell CKA in preliminary calibration, then fixed; Broad and Cardiac scores did not enter this selection. The earlier eight-loss pilot runs selected the internal coefficients in Equation 6; those values remain fixed in the main models and N/W controls.

Let f be the fraction of the record budget processed before an update. During $0 \le f < 0.05$, the objective is

$$J_{\text{GT}} + m(f)\big[J_{\text{expr}} + 0.5J_{\text{cell}} + 0.5J_{\text{internal}} + 0.05J_{\text{gene}}\big], \qquad m(f) = 0.25 + 0.75\frac{f}{0.05}. \tag{12}$$

The middle stage weights the five objectives with Nash-MTL; the final stage uses J_{GT} alone. In the main runs, the stages cover updates 1–1,286, 1,287–23,134, and 23,135–25,704. Learning-rate warm-up instead lasts 1,285 updates, followed by cosine decay to zero. The one-update difference follows from the record-based phase boundaries. Incomplete final batches are retained. Training assigns mask ratios from $\{0.25, 0.5, 0.75\}$ to groups of eight consecutive cells, independently of cell types. Main training uses Python 3.10.20, PyTorch 2.5.1+cu121, CUDA 12.1, and FlashAttention-2 2.8.3.

The Gram matrix uses shared gene/value-encoder and Transformer parameters, with zeros for missing derivatives. Damped Newton solves Equation 3 every 20 updates, maintaining positive weights, with at most 200 iterations and tolerance 10^{-5}. No sum or mean normalization of weights is applied. When a finite positive solution exists, the direction $v = -\sum_{j=1}^{5}\alpha_j \boldsymbol{g}_j$ satisfies

$$D_v J_i = \boldsymbol{g}_i^\top v = -(Q\alpha)_i = -\alpha_i^{-1} < 0. \tag{13}$$

This is a local shared-gradient property (Navon et al., 2022), not a guarantee for each finite AdamW step, which also includes momentum, preconditioning, weight decay, and clipping.

Scope. scPACK requires corresponding gene identities, prediction tasks, and accessible features. Other teachers may need different prediction losses, feature mappings, or schedules. The experiments cover one teacher family with one seed per configuration. Before LayerNorm, $\operatorname{rank}(AD) \le \min(G, K, d)$, with row sparsity further constraining the coefficients. No parameter-matched low-rank comparison isolates SLD's effect. Deployment compression does not establish how small the teacher could have been during pretraining.

B Data and Evaluation

B.1 Data Selection and Evaluation Roles

The local corpus was built from CELLxGENE Discover dataset revisions rather than from a fixed archived Census snapshot. Training uses 6,580,200 records from 450 releases and 106 collections; 3,748,936 are marked primary and 2,831,264 are not. We use May 8, 2023, at 23:59:59 UTC as the reference cutoff, matching the CELLxGENE Census version recorded in scGPT's official preprocessing configuration; the scGPT paper reports the May 15, 2023 Census release (Cui et al., 2024). Training releases have `published_at` on or before the cutoff, and Broad and Cardiac use later CELLxGENE releases. This criterion concerns source-release metadata, not paper publication years. Training, Broad, and Cardiac have no record, release, or collection identifiers in common.

ID contains 19,800 record-held-out profiles from 437 releases and shares 3,824 dataset-scoped donors with training. It is used for preliminary calibration, not as an untouched donor-disjoint test. Broad spans 172 releases and 500 labels overall. Its support/validation/query partitions contain 471/394/410 labels and 1,520/485/488 dataset-scoped donors. Twenty query labels, covering 468 profiles, are absent from support and remain in scoring.

Cardiac is the selected myocarditis myocardial-nucleus subset from the CELLxGENE release of Maatz et al. (2025), not the full published dataset. Its 24 donors are split 14/5/5 across support/validation/query, with nine labels in each. A dataset-scoped donor ID combines a dataset identifier and donor label; these IDs are disjoint across partitions but may not identify unique people across all studies.

Neither Broad nor Cardiac selects the current compression architecture or distillation hyperparameters; Cardiac was blind during those choices. Downstream support fitting and validation checkpoint selection do not feed back into compression design. Query is used only for scoring. Study composition and label granularity differ between the datasets.

B.2 Model States, Inputs, and Classifiers

All evaluations except FT use the base teacher and distilled students before task-specific adaptation. FT produces a separate backbone and classifier per dataset. N/W/G are independent smaller-budget runs, not slices of main-student training. Table 5 gives each evaluation set.

Table 5: Evaluation sample sets. Cell CKA and reconstruction include support, validation, and query; annotation scores use query only.

Analysis	ID	Broad	Cardiac
Global cell CKA / reconstruction	19,800	120,000	62,705
Local cell neighbors	15,000	15,000	–
Annotation / few-shot query	–	20,563	13,721
Clustering / donor mixing	–	5,000	5,000
Fixed-checkpoint objectives	1,024	–	–

Inputs. The model uses positive raw counts, without library-size normalization or log transformation. For each cell, we compute 50 empirical quantile boundaries at evenly spaced levels from zero to one over all nonzero counts. Digitization assigns levels 1–50 under the `n_bins=51` convention; repeated boundaries use a randomized tie rule seeded by the cell's unique record identifier (UID). Zero-count genes are omitted. After binning, at most 1,199 genes are sampled uniformly and CLS is prepended. Padding and masking use -2 and -1, not expression levels. Masking changes the input values, not gene IDs; the original bin values remain the GT targets. Input selection and evaluation masks use the same UID-derived seeds across models.

Frozen classifiers and fine-tuning. kNN uses cosine similarity of row-normalized features, $k = 15$, and uniform votes. Class-vote ties use lexicographic order; neighbor ties use support-row order. LP uses support-fitted standardization and logistic regression with $C = 1$, balanced class weights,

LBFGS, and tolerance 10^{-4}. Fits are retried with iteration caps 300/1,000/3,000/10,000 until they finish without a convergence warning and before the cap; query scores do not select the cap.

FT updates the backbone and a new linear head for five epochs using unweighted cross-entropy. AdamW uses backbone/head learning rates $2 \times 10^{-5}/10^{-3}$, weight decay 0.01, batch size eight, bfloat16, clipping norm one, and seed 20260824. Validation Macro-F1 selects the checkpoint, with the earlier epoch chosen in a tie. Unlike LP, FT changes the backbone and uses equal example weights. This difference is consistent with higher Broad accuracy but similar or lower Macro-F1; it does not isolate class weighting as the cause.

Expression baseline. Each profile is normalized to library size 10,000 and log-transformed with $\log(1+x)$. The 2,000 highest-variance genes and 50 centered PCA components are fitted on support only; validation and query are transformed with those fits. This PCA is distinct from the teacher-embedding PCA used for initialization. Splits and labels match the neural models, but input selection and preprocessing differ. PCA has kNN, LP, and clustering results, not an FT model.

B.3 Annotation, Few-Shot, and Clustering Metrics

For the fixed query label set $\mathcal{C}$ (410 Broad or nine Cardiac labels),

$$\mathrm{MacroF1} = \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} \frac{2\,\mathrm{TP}_c}{2\,\mathrm{TP}_c + \mathrm{FP}_c + \mathrm{FN}_c}. \tag{14}$$

TP, FP, and FN count true positives, false positives, and false negatives. A zero denominator contributes zero; predictions outside $\mathcal{C}$ remain errors on their true classes. Validation uses its own fixed observed label set. Accuracy is the fraction of correctly labeled profiles.

Few-shot kNN uses nominal fractions $f \in \{0.01, 0.05, 0.10, 0.25, 0.50, 1\}$. Each class with n_c support profiles contributes $\max(1, \lceil f n_c \rceil)$ samples. Seeds are 20260841–43, with shared sampled UIDs across models; fractions are drawn separately and need not be nested. Actual support counts are 1,096/4,114/7,934/19,454/38,640/77,018 on Broad and 368/1,825/3,645/9,106/18,206/36,406 on Cardiac. Per draw,

$$\mathrm{AUC}_{\log} = \frac{\mathrm{trapz}(\mathrm{MacroF1}(f), \log f)}{\log(1) - \log(0.01)}, \tag{15}$$

where trapz applies the trapezoidal rule to the six points. This summarizes label efficiency, not ROC area. The plotted ranges span three support draws, not independent distillation seeds.

Clustering uses KMeans with Euclidean distance, ten initializations, and seed 20260824 on the common 5,000-profile query subset. Cluster counts equal its observed cell-type counts: 327 for Broad and nine for Cardiac. NMI and ARI compare assignments with cell types. ASW-cell averages cosine silhouettes by cell type, retaining negative values. For donor mixing, let $\sigma_i^{(c)}$ be profile i's donor silhouette within cell type c, with n_c profiles:

$$M_c = 1 - \left| \frac{1}{n_c} \sum_{i=1}^{n_c} \sigma_i^{(c)} \right|, \qquad \mathrm{ASW}_{\mathrm{donor}} = \frac{1}{|\mathcal{E}|} \sum_{c \in \mathcal{E}} M_c. \tag{16}$$

Types are eligible when $1 < \#\mathrm{donors} < n_c$. Broad has 188 eligible types and excludes 139 types containing 422 profiles; Cardiac has eight eligible types and excludes one single-profile type. Both have $|\mathcal{E}| > 0$. The mean is taken before the absolute value; this is a within-type donor proxy, not the full scIB score.

B.4 Representation Similarity and Reconstruction

For column-centered teacher and student matrices $X \in \mathbb{R}^{n \times 512}, Y \in \mathbb{R}^{n \times d}$ containing the same objects in the same order, linear CKA is (Kornblith et al., 2019)

$$\mathrm{CKA}(X, Y) = \frac{\|X^\top Y\|_F^2}{\|X^\top X\|_F \|Y^\top Y\|_F}, \tag{17}$$

with nonzero denominator. Global cell CKA uses full sets. Local analysis uses common 15,000-profile ID/Broad subsets; on those same subsets, CKA is 0.8643/0.8859 for ID and 0.8749/0.8788 for Broad

(S8–384/S6–256). No Cardiac local-cell result is reported. Gene analysis uses all 60,694 ordinary genes after gene-encoder LayerNorm, before adding expression-value encodings and entering the Transformer, without adapters.

For integer $1 \le k \le n-1$, let $N_T^k(i), N_S^k(i)$ contain the k nearest neighbors by cosine similarity, excluding object i. We average

$$R_i^k = \frac{|N_T^k(i) \cap N_S^k(i)|}{k}, \qquad J_i^k = \frac{|N_T^k(i) \cap N_S^k(i)|}{|N_T^k(i) \cup N_S^k(i)|} \tag{18}$$

over objects to obtain neighbor Recall@k and Jaccard@k. The nonlinear relation $J_i^k = R_i^k/(2 - R_i^k)$ does not hold between their means.

Reconstruction uses mask ratio 0.5 on each full evaluation set, with identical selected inputs and masks across models. Squared errors are pooled over 10,088,056 masked positions on ID, 63,175,670 on Broad, and 33,385,660 on Cardiac. GEP/GEPC predictions are real-valued; targets are the unmasked within-cell bin values, not raw counts. These errors compare task predictions with ground-truth bin values; J_{expr} instead compares student and teacher predictions.

C Additional Task Results

C.1 Clustering, Reconstruction, and Label Efficiency

Tables 6–8 give clustering, reconstruction, and label-efficiency summaries. At 100% support, all three few-shot draws use the same reference set.

Table 6: Clustering and donor mixing on shared query subsets. Definitions of NMI, ARI and ASW are in Appendix B.3.

Dataset	Model	NMI	ARI	ASW-cell	ASW-donor
Broad	PCA	0.7464	0.2239	-0.3187	0.7769
Cardiac	PCA	0.7965	0.7701	0.3128	0.9233
Broad	Teacher	0.7562	0.2199	-0.1326	0.7992
Cardiac	Teacher	0.8263	0.7643	0.4264	0.9321
Broad	S6–256	0.7447	0.2129	-0.1741	0.7994
Cardiac	S6–256	0.8195	0.7863	0.4435	0.9195
Broad	S8–384	0.7489	0.2089	-0.1483	0.7993
Cardiac	S8–384	0.8363	0.7977	0.4808	0.9194

Table 7: **Masked expression reconstruction.** MSE on binned expression levels, pooled over masked positions in each full evaluation set; lower is better.

Dataset	Model	GEP	GEPC
ID	Teacher	222.67	176.39
ID	S8–384	142.34	143.53
ID	S6–256	142.98	143.68
Broad	Teacher	222.86	181.71
Broad	S8–384	151.12	152.33
Broad	S6–256	151.68	152.39
Cardiac	Teacher	225.09	182.67
Cardiac	S8–384	172.34	173.67
Cardiac	S6–256	173.06	173.69

Table 8: Label-efficiency AUC over log support fraction: mean and range of the three support draws.

Dataset	Model	Mean	Minimum	Maximum
Broad	Teacher	0.2220	0.2215	0.2225
Broad	S8–384	0.2057	0.2045	0.2065
Broad	S6–256	0.1965	0.1948	0.1977
Cardiac	Teacher	0.7746	0.7656	0.7877
Cardiac	S8–384	0.7828	0.7786	0.7885
Cardiac	S6–256	0.7802	0.7729	0.7873

C.2 Sensitivity to Donor Composition

We assess sensitivity of Macro-F1 to the observed donor composition using 1,000 draws of independent positive $\mathrm{Exp}(1)$ donor weights, with seed 20260921. Draws share weights across models and keep every query class represented. Scores and paired differences use the weighted confusion counts. Table 9 reports the central 95% ranges for both W–G and N–W.

These ranges condition on the observed donors and trained models; they are not confidence intervals with verified population coverage or variation across training seeds. Nonlinear Macro-F1 averaging can shift their centers away from the full-query score. Broad query contains 488 dataset-scoped donors; Cardiac query contains five. Main-model reconstruction scores are point estimates without per-cell uncertainty estimates.

Table 9: Macro-F1 differences and central 95% positive-donor-weight sensitivity ranges. W–G assesses supervision; N–W assesses dynamic weighting. These condition on single training runs.

Dataset	Readout	W–G: point [range]	N–W: point [range]
Broad	kNN	$+0.0539$ $[+0.0464, +0.0586]$	-0.0036 $[-0.0072, +0.0009]$
Broad	LP	$+0.0598$ $[+0.0507, +0.0674]$	-0.0142 $[-0.0195, -0.0070]$
Cardiac	kNN	$+0.1058$ $[+0.0892, +0.1194]$	-0.0129 $[-0.0215, -0.0077]$
Cardiac	LP	$+0.0112$ $[+0.0063, +0.0155]$	-0.0026 $[-0.0074, +0.0051]$

D Ablation Results and Training Analysis

D.1 Matched-Budget Training and Downstream Results

N/W/G each process 1,645,050 records in 6,426 updates, sharing the S6–256 architecture, sparse initializer, seed 20260824, and input order. Effective batch size is 256 with eight-cell microbatches; the last update has 250 profiles. AdamW uses learning rate 5×10^{-4}, betas $(0.9, 0.999)$, weight decay 0.01, and clipping norm one. Learning-rate warm-up is 321 updates, followed by cosine decay.

N/W warm-up covers updates 1–322, multi-objective training 323–5,784, and GT-only training 5,785–6,426. Their post-warm-up model, adapter, optimizer, and random states match, as does their complete batch order. W solves once at the first multi-objective update and holds the weights fixed; N refreshes every 20 updates, for 274 solves. W's weights in GT/expr/cell/internal/gene order are

$$(0.006291,\ 0.004701,\ 7.399843,\ 5.505478,\ 43.897750).$$

G uses GT throughout, with no teacher forward or Nash updates, but keeps teacher-derived initialization. W–G compares the teacher-supervised and GT-only recipes, including their different stages; it does not isolate one KD term.

Tables 10–11 retain all evaluated downstream measures. N and W improve annotation over G. W has higher final kNN and LP point estimates than N on both datasets, and lower Cardiac reconstruction errors. The N–W LP differences are -0.0142 on Broad and -0.0026 on Cardiac. Matching the teacher's expression predictions during training and final downstream performance are therefore distinct outcomes.

Table 10: Annotation and clustering for N/W/G, with one training seed and the same data budget. kNN/LP values are Macro-F1.

Dataset	Variant	kNN	LP	NMI	ARI
Broad	N	0.2211	0.3709	0.6881	0.1494
Cardiac	N	0.8051	0.7701	0.6975	0.5477
Broad	W	0.2246	0.3851	0.6937	0.1526
Cardiac	W	0.8180	0.7727	0.7059	0.5952
Broad	G	0.1707	0.3253	0.6407	0.1101
Cardiac	G	0.7122	0.7614	0.5754	0.4210

Table 11: Donor mixing and GT reconstruction for N/W/G. Broad reconstruction was not evaluated for these variants.

Dataset	Variant	ASW-cell	ASW-donor	GEP MSE	GEPC MSE
Broad	N	-0.3637	0.7761	–	–
Cardiac	N	0.2944	0.9229	176.59	178.55
Broad	W	-0.3338	0.7859	–	–
Cardiac	W	0.2968	0.9248	175.66	178.17
Broad	G	-0.5208	0.7400	–	–
Cardiac	G	0.0856	0.8866	175.83	178.63

D.2 Training Losses and Fixed-Input Checkpoints

Table 12 averages the five unweighted objectives over 5,462 paired multi-objective updates. Figure 5b,c shows means of ten fixed 500-update blocks and a final 462-update block. GT-only training does not log KD losses; missing values are not zeros.

Table 12: Mean unweighted training losses over 5,462 matched updates; J_{expr} matches teacher GEP/GEPC predictions.

Model	J_{GT}	J_{expr}	J_{cell}	J_{internal}	J_{gene}
N	357.867297	25.128614	0.038150	0.122382	0.390501
W	329.218622	44.630422	0.037650	0.120709	0.390445

Table 13: **GT consolidation.** Unweighted losses on 1,024 identical ID profiles. J_{expr} matches teacher GEP/GEPC predictions. Labels 90/100 mark checkpoints before/after consolidation; lower is better.

Checkpoint	J_{GT}	J_{expr}	J_{cell}	J_{internal}	J_{gene}
N90	340.7799	13.7247	0.0267	0.1039	0.3895
W90	310.3838	36.7358	0.0254	0.1032	0.3895
N100	293.7938	97.5314	0.0318	0.1081	0.3896
W100	292.7353	98.0957	0.0266	0.1050	0.3896

The checkpoint comparison uses steps 5,784 and 6,426, labeled 90/100, on 1,024 identical ID profiles in 128 batches of eight. Each checkpoint uses its own trained adapters and the same inputs, masks, and selected token IDs (cap 4,096, diagnostic step zero). Models and adapters are fixed in evaluation mode. The comparison uses one RTX 5090 Laptop, Python 3.10.12, PyTorch 2.8.0+cu128, CUDA 12.8, and FlashAttention-2 2.8.3, with bfloat16 autocast and float loss reductions. It supplies no hardware-efficiency result.

GT-only consolidation lowers both models' GT reconstruction loss but increases their prediction-matching loss J_{expr}: the W–N gap changes from 23.0111 to 0.5644. Final cell, internal, and gene losses also remain higher for N. This comparison does not measure the stage's effect on downstream scores or establish an optimal schedule.

D.3 Gradient Conflict and Weight Choice

At N's 274 recorded states, GT/expr gradient cosine $Q_{ij}/\sqrt{Q_{ii}Q_{jj}}$ is negative in 264 cases (96.35%), with median -0.9427. The denominator is positive for these gradients. Applying dynamic α_t or initial fixed α_0 to each same Q gives all five entries of $Q\alpha$ positive in 274 or 36 cases, respectively. The fixed weights give a negative expr entry in 225 cases. This illustrates the local direction in Equation 13, not W's actual training failure rate or equality of objective contributions. W has only its initial Gram measurement. The largest recorded Nash residual is below 9.62×10^{-6}. During multi-objective training, clipping occurs at every N update and 56.35% of W updates, so actual optimizer steps also differ in scale.

E Inference and Parameter Accounting

Table 14 gives all inference measurements under Section 4's matched conditions: isolated RTX A6000 processes, 1,200-token inputs, bfloat16, FlashAttention-2, and cell/GEP/GEPC outputs. Fifty warm-ups precede 200 synchronized CUDA timings. Throughput is batch size divided by mean latency; loading, transfer, energy, and maximum feasible batch size are not measured.

Table 14: All A6000 measurements. Alloc./res. are peak PyTorch allocated/reserved memory in GiB; latency is per batch.

Model	Batch	Cells/s	Latency (ms)	Alloc.	Res.
Teacher	1	35.37	28.27	0.243	0.271
Teacher	8	220.97	36.20	0.368	0.449
Teacher	32	387.82	82.51	0.805	1.174
Teacher	64	444.30	144.05	1.381	2.145
S8–384	1	57.45	17.41	0.125	0.176
S8–384	8	365.28	21.90	0.526	0.734
S8–384	32	636.40	50.28	1.893	2.768
S8–384	64	612.13	104.55	3.718	5.480
S6–256	1	76.01	13.16	0.086	0.109
S6–256	8	530.66	15.08	0.353	0.477
S6–256	32	1091.37	29.32	1.269	1.854
S6–256	64	1271.75	50.32	2.491	3.668

The teacher has 51,329,025 parameters used by the shared cell/GEP/GEPC functions. Its original checkpoint has 51,330,049 floating-point entries; an unused 1,024-parameter flag embedding is excluded, while the runtime class constructs an unused 527,873-parameter classification head:

$$51{,}330{,}049 - 1{,}024 + 527{,}873 = 51{,}856{,}898.$$

This explains the approximately 51.86M instantiated count. The classification head is not the CLS pooling token and is not executed by the benchmark. Parameter ratios exclude the unused head; file sizes measure actual saved bytes.

The dense teacher gene embedding table contains $60{,}697 \cdot 512 = 31{,}076{,}864$ values. The student allocates Vs coefficients, Kd atom values, and $3d$ special-token values. For S6–256, these total $971{,}152 + 65{,}536 + 768 = 1{,}037{,}456$ floats before gene LayerNorm. Three special-token coefficient rows are allocated but bypassed. The fixed int16 indices occupy $60{,}697 \cdot 16 \cdot 2 = 1{,}942{,}304$ bytes. Three adapters add $3 \cdot 512 \cdot d$ training parameters (589,824 or 393,216) and are excluded from deployment.

File sizes use $1\ \mathrm{MiB} = 2^{20}$ bytes; memory describes the PyTorch allocator. Sparse lookup gathers a temporary $[B, T, s, d]$ tensor before summing, consistent with high-batch memory use but not an operator-level attribution. Parameter count, precision, storage, and runtime memory are separate measures.

F Qualitative Analysis

F.1 Sparse Gene-to-Atom Structure

Each main student uses all 256 atoms and 16 nonzero coefficients for each of 60,694 genes, giving 971,104 gene-to-atom connections. For selected slot j of gene g, we summarize component magnitudes before LayerNorm as

$$q_{gj} = |c_{gj}| \, \|D_{I_{gj},:}\|_2, \qquad \rho_{gj} = \frac{q_{gj}}{\sum_{j'=1}^{s_g} q_{gj'}}, \qquad \sum_{j'=1}^{s_g} q_{gj'} > 0. \tag{19}$$

The median largest-component share is 0.1067/0.1052 for S8–384/S6–256; the top-three shares are 0.2864/0.2825. Scaling an atom and inversely scaling its coefficients preserves q_{gj}. The proxy ignores cancellation and downstream processing and is not a causal attribution. Table 15 retains examples from the same four fixed atom IDs; these atoms are not identified as validated biological regulators.

Table 15: Two highest-ranked genes from each of the four fixed atoms in S6–256, ranked by q_{gj}. Numbers in parentheses are signed coefficients, not attributions. Atom IDs are zero-based.

Atom	First gene (coefficient)	Second gene (coefficient)
0	RBM39 (+1.7683)	FUS (+1.3139)
64	ZSCAN31 (+1.3425)	MAST3 (+0.4978)
128	ODR4 (+1.3595)	GOSR1 (+0.6436)
192	RP11-535A15.2 (+1.0283)	RP11-569G13.3 (+0.5195)

F.2 Cardiac Cell-Type Visualization

Figure 6 shows the teacher and main students using uniform manifold approximation and projection (UMAP) (McInnes et al., 2018), colored by nine broad cell types. Cell-type groups remain visible, with different arrangements; the plot is qualitative, not a teacher-fidelity metric.

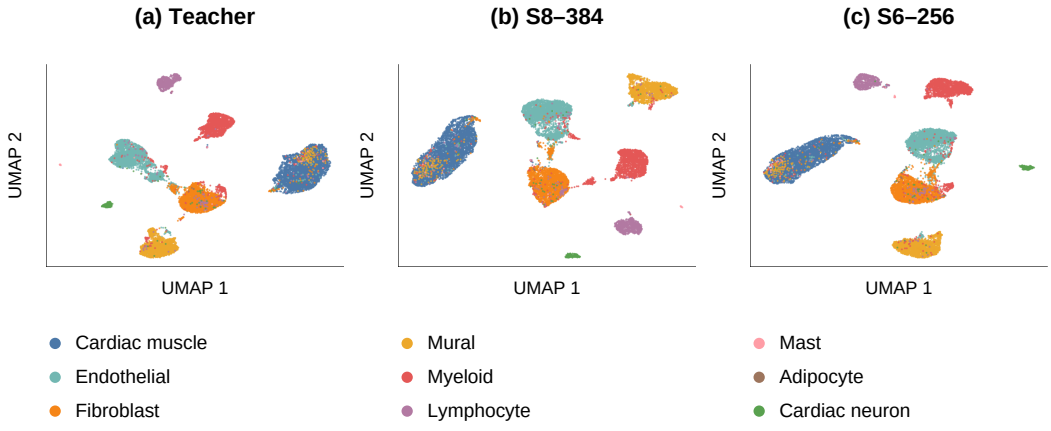


Figure 6: **Cardiac cell-type UMAPs.** Point positions and colors are unchanged from the original visualization. Axes are panel-specific, not a shared teacher–student coordinate system. Per-cell coordinates, selected record identifiers, and the fitted projection are unavailable for this figure; it is not identified with the 5,000-query-profile metric subset. All quantitative clustering and neighborhood results use the high-dimensional features in Appendix B.