BATCH-ROBUST CELL REPRESENTATIONS BY SELF-DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Zero-shot embeddings from masked-reconstruction single-cell foundation models such as scGPT and Geneformer do not reliably beat HVG+PCA at batch integration. Self-distillation methods such as DINO, which optimise the embedding space directly, lead imaging benchmarks but have not yet shown clear gains on scRNA-seq. We train scRAPTOR (single-cell RAPid Teacher-student Optimisation of Representations), a 7.6M-parameter attention-free encoder, by prototype self-distillation on a public corpus of 86 million cells from 1,065 human studies. Sinkhorn equipartition keeps the prototype head from collapsing, and a low-rank study term on the prototype logits, discarded after training, absorbs study identity, so the encoder needs no batch label at inference. Zero-shot, scRAPTOR beats transductive HVG+PCA on scIB Total on the Pancreas, PBMC and DKD benchmarks, and leads every other zero-shot embedding we scored (scGPT, Geneformer, scJEPA, Concerto and two scVI models trained on our corpus) on Pancreas, PBMC and the three-dataset mean. Without Sinkhorn equipartition the head collapses to a dozen prototypes and loses held-out-study accuracy; without the study term the embedding carries more study identity and scores lower on PBMC and DKD. We release the weights of all three seeds, the cell-level split and the atlas build scripts.

1 INTRODUCTION

The dominant design for single-cell foundation models takes masked language models and applies them to expression profiles. For example, scGPT (Cui et al., 2024), Geneformer (Theodoris et al., 2023) and UCE (Rosen et al., 2026) all train transformers to predict masked gene expression (values, ranks or whether a gene is expressed), cell by cell, using learnt gene-level representations. Kedzierska et al. (2025), evaluating Geneformer and scGPT without fine-tuning, i.e. in a zero-shot setting, found that neither reliably outperforms the raw highly variable genes (HVGs), or scVI and Harmony (Korsunsky et al., 2019) fitted on each dataset, on cell-type clustering or batch correction. Two recent joint-embedding models, Cell-JEPA (ElSheikh et al., 2026) and GeneJEPA (Litman et al., 2025), share our premise, argued below, that reconstruction rewards encoding the assay, but neither runs an HVG or PCA baseline: both compare against scGPT, the model subject to the critique. Whether any self-supervised model beats these classical baselines zero-shot therefore remains open; this paper tests it on batch integration.

In scRNA-seq most entries are zero, largely through capture and sampling depth rather than absence of the transcript (Svensson, 2020). The value at a masked gene is therefore dominated by these technical factors, and an objective that rewards predicting it partly rewards encoding the assay. In vision, joint-embedding methods (DINO (Caron et al., 2021), DINOv2 (Oquab et al., 2024)) yield strong frozen features by matching a student's prototype assignment to a momentum teacher's under augmentations chosen for the invariances wanted. The corresponding move for single-cell data is to treat the set of genes captured as an augmentation, so that the model learns invariance to capture instead of predicting it.

Self-distillation has no batch mechanism by construction: every positive pair is two views of one cell, so the pair shares its study, donor and assay, and no ground truth exists to say whether two cells from different batches are the same or not. We therefore give study identity an explicit, discardable home: a low-rank term on the prototype logits, used only in training. Without a usage constraint the prototype head also collapses onto a few prototypes, so we keep them in use with Sinkhorn

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.

equipartition. Together, the two let prototype self-distillation train stably on 86M cells and transfer to unseen batches without a batch label.

Contributions.

1. **Zero-shot embeddings that beat simple baselines.** A 7.6M-parameter encoder trained on public data (mean of three seeds) beats transductive HVG+PCA and raw HVG on scIB Total on Pancreas, PBMC and DKD, and every other zero-shot embedding we scored (scGPT, Geneformer, scJEPA, Concerto, two scVI models) on Pancreas and PBMC; on DKD it is within 0.02 of the best (§5.1).
2. **Batch robustness without a batch label.** A low-rank study term on the prototype logits, discarded after training, leaves in-corpus accuracy unchanged, adds 0.01–0.02 zero-shot Total on PBMC and DKD and lowers linear batch decodability by 0.11–0.17 in single-run ablations; attacking batch identity directly with a donor adversary costs accuracy instead (§5.2, Appendix B).
3. **Sinkhorn equipartition makes prototype self-distillation train at scale.** Without it the head collapses to a dozen prototypes within 20M cells and loses held-out-study accuracy, on our trunk and on Concerto's; with it, training is stable to 172M cells with over 300 prototypes in use, and the prototype-geometry terms are not needed (§5.2).
4. **Release.** Weights of all three seeds, training code, the cell-level split and a build script for the 86M-cell public corpus.

2 Related work

Joint-embedding models for single-cell data. Concerto (Yang et al., 2022), an early teacher–student model for single-cell data, showed that self-distillation works on scRNA-seq with a student that is a single dense layer on the count vector, but was trained per atlas rather than applied zero-shot. Cell-JEPA (ElSheikh et al., 2026) and scJEPA (Zhai et al., 2026) add latent prediction to masked reconstruction, and scConcept (Bahrami et al., 2025) replaces reconstruction with cell-level contrast. Closest to our work are GeneJEPA (Litman et al., 2025) and BioM-JEPA (Wang et al., 2026). Like scRAPTOR, both train a student against a momentum teacher without reconstructing expression, but their student regresses the teacher's representation of genes hidden from it: a random $\sim$45% of each cell's expressed genes in GeneJEPA, and blocks of genes connected in a STRING and co-expression graph in BioM-JEPA. GeneJEPA trains a Perceiver on Tahoe-100M, a drug-perturbation atlas of cancer cell lines (Zhang et al., 2025); BioM-JEPA, a linear-attention transformer on scBaseCount (Youngblut et al., 2026). Both evaluate frozen embeddings with supervised probes for cell-type annotation and perturbation or drug response, and neither reports batch integration or an HVG+PCA baseline. Unlike all of these, scRAPTOR asks two gene-dropout views of a cell to agree on an assignment over learned prototypes rather than regressing or contrasting a latent, and prevents collapse by equipartitioning prototype usage rather than penalising prediction variance and covariance. It also adds a discardable study term where GeneJEPA and BioM-JEPA have no batch mechanism (a gap GeneJEPA acknowledges). Finally, it is evaluated zero-shot against the raw-HVG baseline of the critique (Kedzierska et al., 2025) and against HVG+PCA; the critique's fitted scVI and Harmony learn from each evaluation set's batch labels, so we report fitted integrators on the immune atlas (Table 2) and compare inference-time embeddings on the benchmarks.

Prototype assignment, collapse and batch terms. Prototype self-distillation needs a guard against collapse: the student only matches the teacher's assignment, so a teacher that uses few prototypes gives the trunk few groups to separate, far too few for a corpus whose cell-type abundances span orders of magnitude. SwAV's Sinkhorn–Knopp equipartition (Caron et al., 2020), which DINOv2 adopted in place of centring (Oquab et al., 2024), constrains how often prototypes are used and so prevents collapse onto a few prototypes, though not the near-duplicate prototypes of partial collapse (Govindarajan et al., 2024; Arteaga et al., 2026). Per-group correction terms are standard outside self-distillation, in mixed-effects deep learning (Nguyen et al., 2023), logit adjustment (Menon et al., 2021) and scVI's batch-conditioned decoder (Lopez et al., 2018), which BA-scVI extends with batch adversaries on the encoder and decoder (Díaz-Mejía et al., 2025). The study term is closest to the product-of-experts bias branches of RUBi (Cadene et al., 2019), Clark et al. (2019) and He

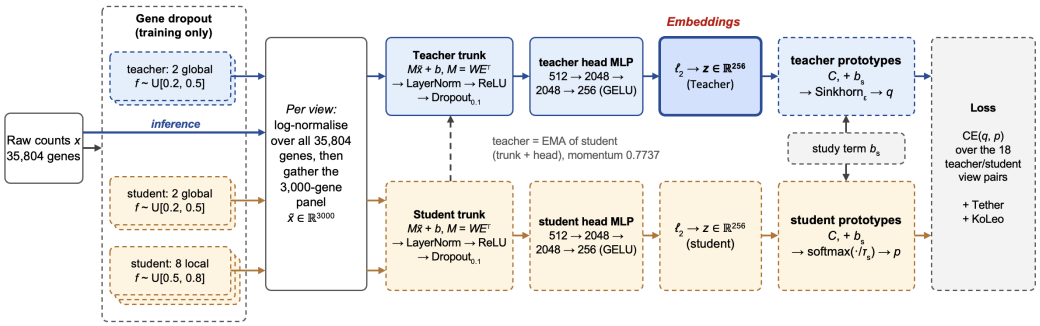

Figure 1: **scRAPTOR and where the embedding is taken.** Blue: teacher; orange: student; solid: used at inference; dashed: training only. The trunk is one linear map $M = WE^\top$ followed by LayerNorm and ReLU; the study term b_s enters only the training-time prototype logits. The teacher's ℓ_2-normalised 256-d head bottleneck is the embedding.

et al. (2019), which are likewise dropped at test time. Conditional contrastive learning (Tsai et al., 2022) targets the same nuisance in contrastive learning, and DESC (Li et al., 2020), which self-trains on sharpened cluster assignments and removes batch effects in the process, is the closest single-cell precursor. scRAPTOR combines both ideas in one head: equipartition against collapse, and a bias branch dropped at test time for study identity.

3 METHOD

scRAPTOR is an adapted DINO self-distillation model. A student and an exponential-moving-average teacher each map a cell to a distribution over $K = 1{,}024$ prototypes, and the student is trained to match the teacher's assignment on a different view of the same cell. The teacher's trunk and head MLP are kept; the prototypes and study term are used in training alone (Figure 1; all settings in Table 4, Appendix A).

Input and trunk. A cell is its raw count vector x over the harmonised 35,804-gene vocabulary. A view applies gene dropout to x (see *Views and teacher*). The result is then library-size normalised and log-transformed over the full vocabulary, $\tilde{x} = \log(1 + 10^4\, x / \sum_g x_g)$, and only then gathered to a fixed 3,000-gene HVG panel, so the encoder reads $\tilde{x} \in \mathbb{R}^{3000}$ (Figure 1). Gathering first would renormalise each cell against a biased, subset-dependent library size. The trunk is a gene-embedding table $E \in \mathbb{R}^{3000\times512}$ (1.5M parameters), summed over the cell's detected genes and weighted by the normalised value, followed by one linear layer (263K parameters):

$$h \;=\; \mathrm{Dropout}_{0.1}\big(\mathrm{ReLU}(\mathrm{LayerNorm}(WE^\top\tilde{x} + b))\big), \qquad W \in \mathbb{R}^{512\times512}.$$

Nothing separates E and W, so the trunk is a single linear map $M\tilde{x} + b$, $M = WE^\top \in \mathbb{R}^{512\times3000}$, followed by LayerNorm and ReLU; weight decay on the two factors acts as a low-rank penalty on M. The trunk is thus a linear projection followed by LayerNorm and ReLU, as close to PCA as a learned encoder gets; the head MLP adds depth before the embedding.

Head, study term and assignment. The projection head is a three-layer MLP ($512 \to 2048 \to 2048 \to 256$, GELU), followed by ℓ_2 normalisation and a weight-normalised linear layer C onto $K = 1{,}024$ prototypes. The weight-norm gains are fixed at 1, so every prototype is a unit vector. The embedding is the teacher's ℓ_2-normalised 256-d MLP output (the bottleneck), so the encoder at inference is the trunk plus the MLP, 7.57M parameters; C and the study term are used only during training.

Study identity enters during training as a low-rank offset $b_s = A\,e_s$, added to both the student and teacher prototype logits. Here e_s is a learned 128-d embedding of study s (one of 1,077), and $A \in \mathbb{R}^{1024\times128}$ is a zero-initialised projection, so the offset is zero at the start but still trainable

(0.27M parameters). We call this the study term. The design follows the reasoning behind scVI conditioning its decoder, not its encoder, on batch (Lopez et al., 2018). Any part of the teacher's assignment that can be predicted from study identity is a target the trunk could only hit by encoding the study. Giving that part an explicit, discardable home leaves the trunk to model the study-free residual. Because the study term is discarded with the prototypes, embedding a cell at inference requires only its raw counts, with no study label.

In place of DINO's centring and sharpening, we normalise teacher logits to a uniform marginal over prototypes with Sinkhorn–Knopp (Cuturi, 2013), as in SwAV (Caron et al., 2020) and DINOv2 (Oquab et al., 2024). We use $\varepsilon = 0.05$ and three iterations over the current global batch's teacher logits (16,384 rows) with a queue of the 4,096 preceding ones prepended, and a student temperature $\tau_s = 0.1$. Sinkhorn thus constrains how evenly the prototypes are used, but not how they are arranged.

We therefore add two regularisers on prototype geometry. A tether pulls each student prototype towards its own fixed, randomly chosen hypercube corner, a soft form of the fixed classifier of Hoffer et al. (2018). A KoLeo term (Sablayrolles et al., 2019) pushes each prototype away from its nearest neighbour. We track usage with the perplexity of the mean teacher assignment, $\mathrm{PPL}(\bar{q}) = \exp(-\sum_k \bar{q}_k \log \bar{q}_k)$, where $\bar{q} = \frac{1}{N}\sum_i q_i$; $\mathrm{PPL}(\bar{q})$ equals K when usage is uniform and 1 under total collapse. The ablations (§5.2) show that usage, not geometry, is what matters: the geometry terms are not needed once Sinkhorn is in place.

Views and teacher. The only augmentation is gene dropout. For each view, a rate f is drawn per cell, uniformly from $[0.2, 0.5]$ for a global view and from $[0.5, 0.8]$ for a local view. Each gene is then independently zeroed with probability f on the raw counts, before normalisation. Each cell yields two global teacher views, two global student views and eight local student views. The teacher is an exponential moving average of the student with momentum 0.996 at a reference batch of 128, rescaled to the global batch as $0.996^{64} = 0.7737$ (Busbridge et al., 2023). Because dropout acts on raw counts, the invariance the model learns is invariance to which genes happened to be captured.

Loss and optimisation. Let z_v^s and z_u^t be the ℓ_2-normalised head outputs of student view v and teacher view u of a cell from study s. Let $q_u = \mathrm{Sinkhorn}_\varepsilon\big(Cz_u^t + b_s\big)$ be the equipartitioned teacher assignment (entropic regularisation $\varepsilon = 0.05$ takes the place of a teacher temperature). The loss is

$$\mathcal{L} = \frac{1}{|\mathcal{P}|} \sum_{(u,v)\in\mathcal{P}} \mathrm{CE}\Big(q_u,\ \mathrm{softmax}\big((Cz_v^s + b_s)/\tau_s\big)\Big) \ +\ \lambda_{\mathrm{t}}\,\mathcal{L}_{\mathrm{tether}}(C) \ +\ \lambda_{\mathrm{k}}\,\mathcal{L}_{\mathrm{KoLeo}}(C), \quad (1)$$

the DINO cross-entropy between each of the two teacher assignments and each of the ten student views, over the $|\mathcal{P}| = 18$ pairs that exclude the same-index pair, with $\lambda_{\mathrm{t}} = 0.1$ and $\lambda_{\mathrm{k}} = 0.05$. The optimiser is LAMB (You et al., 2020), a layer-wise adaptive optimiser designed for large global batches, with gradient clipping at 3.0 and weight decay 0.04 on every parameter. The learning rate is 10^{-4} at a reference batch of 128, scaled linearly to 6.4×10^{-3} at the global batch of 8,192, with 2,000 batches (16.4M cells) of linear warm-up and no decay; prototypes are frozen for the first 1,000 batches (8.2M cells). The whole recipe runs on one A10G GPU, and a seed reaches early stop in 13.8–21.1 h (Table 4).

4 Data and evaluation protocol

Corpus. We train on CorpusX, an atlas of 112,845,978 cells across 1,077 public studies. The training split holds 86,112,911 cells from 1,065 studies and 25,029 samples, capped so that no (study, tissue, disease) stratum contributes more than 100k cells. The atlas build package, with a reproduction recipe, and the split file that fixes exactly which cells were trained on are in the supplementary material (Appendix A).

Benchmarks and exposure. We evaluate on the scIB human-pancreas compilation (Luecken et al., 2022) used by the critique (Kedzierska et al., 2025) (Pancreas; 16,382 cells, 9 technology batches), the scSSL-Bench (Ovcharenko et al., 2025) PBMC object of Ding et al. (2020) (30,449 cells, 8 batches) and the Open Problems (Open Problems Consortium, 2026) diabetic kidney dataset of Wilson et al. (2022) (DKD; 39,176 cells). Audited by accession and PubMed identifier, Pancreas and PBMC are

Table 1: **scRAPTOR has the best mean score over Pancreas, PBMC and DKD.** Raw scIB scores (scIB metrics 0.5.1, 4,000 batch-aware HVGs, as in Ovcharenko et al. (2025)) and strict cross-batch kNN accuracy; last columns: means. scRAPTOR is the teacher's 256-d head bottleneck. Our row averages three seeds (listed at the foot); cells average four to eight scoring passes that differ only in the Leiden seed (Geneformer: one). Bold: best per column, excluding the individual seed rows. Transductive rows are fitted on each evaluation set; the controls are our encoder at its training initialisation, an untrained corpus PCA or random projection with our trunk's LayerNorm (LN) and ReLU, and a centred corpus PCA-256 with ℓ_2 (Appendix D).

embedding	Pancreas				PBMC				DKD				mean	
	Bio	Batch	Total	cross kNN	Bio	Batch	Total	cross kNN	Bio	Batch	Total	cross kNN	Total	cross kNN
transductive (fitted on the evaluation set, using its batch key)														
scGPT (per-dataset HVGs)	0.599	0.399	0.519	0.637	0.637	0.378	0.533	0.735	**0.750**	0.462	0.635	0.853	0.562	0.742
raw HVG	0.680	0.440	0.584	0.611	0.540	0.397	0.483	0.622	0.730	0.427	0.609	0.785	0.559	0.673
HVG+PCA	**0.690**	0.394	0.571	0.434	0.565	0.358	0.482	0.413	0.719	0.416	0.598	0.654	0.550	0.500
zero-shot (never sees the evaluation set or its batch labels)														
scRAPTOR	0.663	**0.540**	**0.614**	**0.790**	**0.762**	**0.619**	**0.705**	**0.886**	0.680	0.524	0.618	0.866	**0.646**	**0.847**
scVI on our trunk	0.647	0.461	0.573	0.718	0.719	0.498	0.631	0.878	0.726	0.453	0.617	0.822	0.607	0.806
scVI (our corpus)	0.648	0.398	0.548	0.674	0.673	0.489	0.599	0.855	0.705	0.497	0.622	**0.891**	0.590	0.807
scGPT (scRAPTOR's gene panel)	0.606	0.376	0.514	0.610	0.650	0.506	0.593	0.849	0.746	0.476	**0.638**	0.849	0.582	0.769
scJEPA (public checkpoint)	0.563	0.444	0.515	0.622	0.732	0.474	0.629	0.807	0.640	0.512	0.589	0.824	0.578	0.751
scJEPA (our reproduction)	0.600	0.483	0.553	0.572	0.622	0.419	0.541	0.706	0.601	**0.574**	0.590	0.711	0.561	0.663
Concerto objective (our corpus)	0.492	0.301	0.416	0.486	0.555	0.451	0.513	0.730	0.526	0.392	0.472	0.696	0.467	0.637
Geneformer	0.470	0.290	0.398	0.431	0.478	0.324	0.417	0.591	0.484	0.401	0.451	0.696	0.422	0.573
untrained controls														
random projection-512 + LN + ReLU	0.666	0.475	0.590	0.678	0.654	0.478	0.584	0.827	0.734	0.443	0.618	0.865	0.597	0.790
scRAPTOR encoder at initialisation (trunk + MLP + ℓ_2)	0.625	0.473	0.565	0.708	0.629	0.524	0.587	0.821	0.708	0.498	0.624	0.880	0.592	0.803
corpus PCA-512 + LN + ReLU	0.620	0.413	0.537	0.600	0.678	0.461	0.591	0.814	0.686	0.363	0.557	0.781	0.562	0.732
corpus PCA-256 + ℓ_2	0.626	0.388	0.531	0.589	0.662	0.463	0.582	0.805	0.668	0.331	0.533	0.753	0.549	0.716
scRAPTOR seeds														
seed 20	0.643	0.532	0.598	0.763	0.774	0.610	0.709	0.889	0.674	0.533	0.617	0.870	0.642	0.841
seed 21	0.697	0.542	0.635	0.803	0.742	0.624	0.695	0.883	0.697	0.525	0.628	0.869	0.653	0.852
seed 22	0.650	0.546	0.609	0.804	0.770	0.623	0.711	0.887	0.671	0.514	0.608	0.858	0.643	0.850

absent from our corpus. DKD includes cells of Wilson et al. (2019), whose accession is in the corpus: the 16,155 corpus cells whose barcodes match the DKD object are removed from every split, so no test cell is trained on, and the accession's other cells remain in training (Appendix A). scGPT was likely exposed to the healthy-tissue part of DKD through its CELLxGENE Census snapshot. We also score the Open Problems immune cell atlas (Domínguez Conde et al., 2022), a scSSL-Bench and Open Problems dataset that is in our corpus and in the Census, under both the task's harness and our scIB recipe. Pancreas and PBMC thus test transfer, DKD partial exposure and the immune atlas shared exposure.

Metrics. We score embeddings with scIB (Luecken et al., 2022): Bio conservation (isolated labels, Leiden NMI/ARI, silhouette label, cLISI), Batch correction (silhouette batch, iLISI, kBET (Büttner et al., 2019), graph connectivity, PCR comparison) and Total ($0.6\,\mathrm{Bio} + 0.4\,\mathrm{Batch}$). Scores are raw, not min–max rescaled over the methods in a table, which would make every score depend on which methods are present. On the corpus pools we use $\mathrm{acc}_{\mathrm{cross}}$, a k-nearest-neighbour cell-type vote restricted to neighbours in other batches, to select checkpoints. On the benchmarks we report its strict form ("cross kNN"): each cell votes with the other-batch cells among its 50 nearest neighbours, and a cell with none counts as wrong, so the score rewards both label transfer and mixing. Tables use raw Euclidean distances; for checkpoint selection the same vote runs after scaling each dimension by its interquartile range. Batch decodability is the balanced accuracy of a linear probe for the batch key, averaged over the three benchmarks.

Checkpoint selection. Each run is scored every 2.5M cells on two frozen corpus pools: held-out cells (250,000 cells of the training studies; batch = study, 72 cell types) and held-out studies (355,654 cells of four studies never trained on; batch = sample, 25 cell types). Each pool has a disjoint test half, on which Table 3 reports. Training stops after 20 evaluations without improvement on either pool. The reported checkpoint has the best exactly rescored mean $\mathrm{acc}_{\mathrm{cross}}$ of the two pools among the five best logged evaluations; the learning-rate warm-up is not eligible. The layer is chosen on the same pools: at all 14 candidate checkpoints the 256-d head bottleneck beats the 512-d trunk on both, so we report the head, selected by the same rule. Selection never otherwise touches Pancreas, PBMC, DKD or the immune atlas.

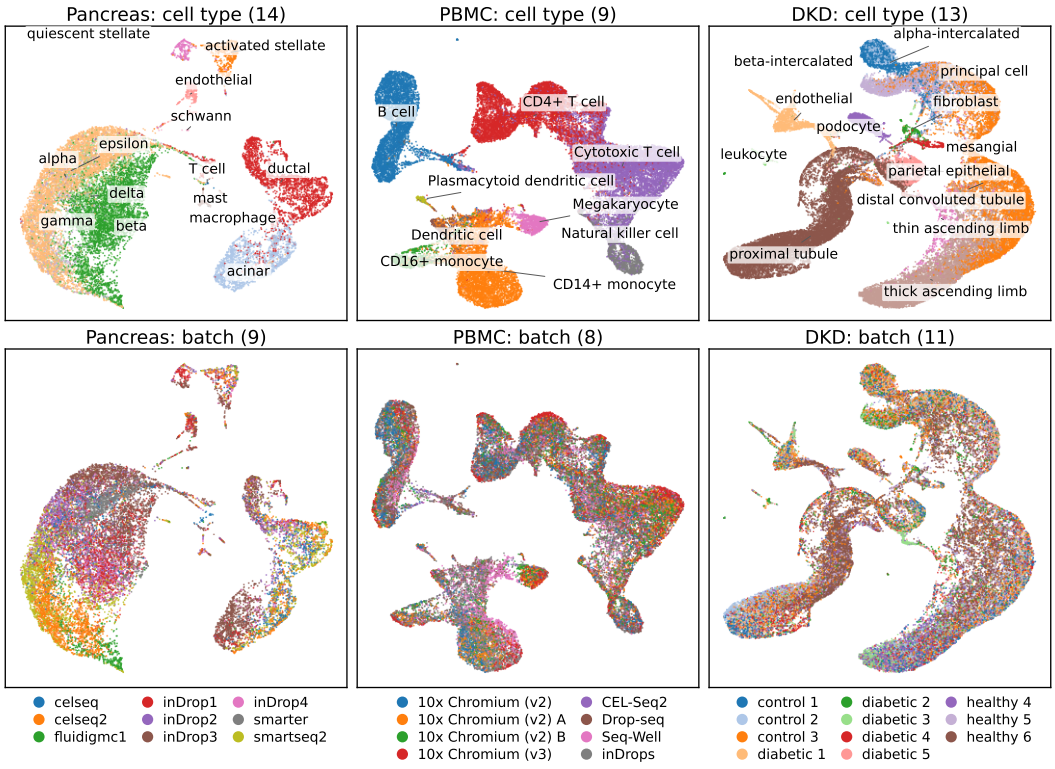

Figure 2: **Zero-shot embeddings.** UMAP of seed 20's embedding, coloured by cell type (top) and batch (bottom); batches interleave within cell types.

5 Results

5.1 Zero-shot batch integration

scRAPTOR has the best mean Total in Table 1 and beats transductive HVG+PCA on every dataset, by $+0.043$, $+0.223$ and $+0.020$; and it beats the critique's baseline, the raw matrix of 2,000 log-normalised HVGs, by $+0.030$, $+0.222$ and $+0.009$ (embeddings in Figure 2). The margins come from iLISI, kBET and PCR, and on Pancreas and PBMC also ARI (Appendix Table 6). PCR, on which HVG+PCA scores low by construction since it is scored against a PCA of the unintegrated data, carries the Pancreas and DKD margins: without it the other nine metrics net to -0.003 and -0.003, against $+0.173$ on PBMC (Appendix Table 7). On strict cross kNN, scRAPTOR has the best mean (0.847, against 0.807 for scVI on our corpus and 0.501 for HVG+PCA, whose unmixed batches leave many cells with no other-batch neighbour). Both simple baselines are thus beaten on Total zero-shot on all three benchmarks, broadly on PBMC and narrowly on Pancreas and DKD.

Among zero-shot embeddings, scRAPTOR leads scGPT, Geneformer, both scJEPA rows, Concerto's objective trained on our corpus and both scVI models on Pancreas and PBMC. Untrained controls isolate the objective. Our own encoder at its training initialisation (trunk, MLP and ℓ_2) trails scRAPTOR by $+0.049$ and $+0.118$ but is slightly ahead of it on DKD (-0.006); linear controls of the trunk's shape trail by $+0.077$, $+0.114$ and $+0.061$ (corpus PCA-512 with LayerNorm and ReLU) and by $+0.024$ and $+0.121$ (random projection, level on DKD), and a corpus PCA-256 with ℓ_2 by $+0.083$, $+0.123$ and $+0.085$. Our lead over the closest, the scVI objective on our own trunk, is $+0.041$ on Pancreas and $+0.074$ on PBMC but only $+0.001$ on DKD.

DKD does not separate the zero-shot methods. scGPT given our gene panel is ahead (0.638 against 0.618, through NMI and ARI), scVI on our corpus is level, and even the random projection and our untrained encoder beat HVG+PCA (0.618 and 0.624 against 0.598); our lead over HVG+PCA rests

Table 2: **scRAPTOR on the in-corpus immune cell atlas.** Open Problems batch-integration score (scaled over the task's 22 published methods) and label-projection (LP) scores, each ranked among methods never exposed to cell-type labels, plus ours (of 14 and 15); label-exposed methods are listed unranked. Bio / Batch / Total: raw scIB axes as the task harness computes them (Table 10 applies Table 1's recipe). Training and exposure: how each method met the atlas and which task metadata it uses. *: LP refitted by us; "–": not refittable (SCA, CellPLM). Our row: mean ± sd of three seeds.

method	training	exposure	BI score	unsup. rank	Bio	Batch	Total	LP accuracy	LP macro-F1	unsup. rank
Pretrained embeddings, no fine-tuning (ours, our corpus-trained scVI models, public scJEPA)										
scRAPTOR	in-corpus	batch	0.579 ±0.007	2	0.731	0.576	0.669	0.874 ±0.005	0.789 ±0.009	1
scVI (our corpus)	in-corpus	batch	0.485	12	0.608	0.450	0.545	0.768	0.641	12
scVI on our trunk	in-corpus	batch	0.481	12	0.605	0.431	0.536	0.824	0.705	7
scJEPA (public checkpoint)	in-corpus	neither	0.521	11	0.595	0.542	0.573	0.847	0.662	5
Published methods (rank over label-free methods only)										
density-adaptive	fit	batch	0.618	1	0.648	0.777	0.700	0.849*	0.691*	3
scANVI	fit	both	0.590	–	0.732	0.557	0.662	0.897	0.810	–
scVI (fitted)	fit	batch	0.557	3	0.666	0.563	0.624	0.874*	0.743*	2
harmonypy	fit	batch	0.550	4	0.624	0.575	0.604	0.789*	0.564*	10
Harmony	fit	batch	0.548	5	0.629	0.575	0.607	0.789*	0.564*	10
pyLIGER	fit	batch	0.544	6	0.590	0.621	0.602	0.717*	0.452*	12
fastMNN	fit	batch	0.532	7	0.630	0.553	0.600	0.831*	0.628*	6
Scanorama	fit	batch	0.526	8	0.622	0.533	0.587	0.849*	0.670*	4
LIGER	fit	batch	0.525	9	0.587	0.622	0.601	0.642*	0.395*	14
ComBat	fit	batch	0.522	10	0.588	0.607	0.596	0.795*	0.561*	9
SCimilarity	in-corpus	ct-label	0.515	–	0.668	0.507	0.603	0.875	0.740	–
scGPT	in-corpus	neither	0.490	11	0.594	0.448	0.535	0.699	0.421	13
CellPLM	neither	neither	0.447	12	0.606	0.406	0.526	–	–	–
no integration	neither	neither	0.410	13	0.578	0.411	0.511	0.800*	0.626*	8
SCA	fit	neither	0.401	14	0.574	0.410	0.508	–	–	–
scANVI + scArches	fit	both	–	–	–	–	–	0.899	0.795	–
logistic regression	fit	ct-label	–	–	–	–	–	0.884	0.775	–
MLP	fit	ct-label	–	–	–	–	–	0.863	0.694	–

on PCR (0.47 against 0.13). On DKD cross kNN we trail scVI on our corpus (0.891) and even our untrained encoder (0.880 against 0.866): random layers blur donor-level structure within each cell type, so almost every cell gains an other-batch neighbour, and DKD's well-separated cell types keep the vote accurate. Where batches are technologies this fails: on Pancreas the untrained encoder falls to 0.708 against our 0.790. On DKD, scRAPTOR is therefore among the best zero-shot embeddings, not ahead of them.

Immune cell atlas. The Open Problems immune-cell-atlas task (Domínguez Conde et al., 2022; 329,762 cells) is in our corpus and, as a CELLxGENE dataset, in most foundation models' pretraining corpora (Table 2, "training"), so it tests shared exposure rather than transfer. Under the task's own harness and without fine-tuning, scRAPTOR reaches a batch-integration score of 0.579 ± 0.007, rank 2 of 14 among methods whose representation never saw a cell-type label. Among these it is above scGPT (0.490), the public scJEPA checkpoint (0.521), CellPLM (Wen et al., 2024), our two corpus-trained scVI models (0.485, 0.481) and every dataset-fitted integrator except the density-adaptive method (0.618), which is fitted on the atlas with its batch key. The ranking over foundation models holds without the isolated-label metric (rank 5). Label projection reaches 0.874 ± 0.005, rank 1 of 15 among label-free representations and level with scVI fitted on the atlas; label-supervised methods, including scANVI (Xu et al., 2021) and SCimilarity (Heimberg et al., 2025), reach 0.86–0.90 and are listed unranked. Under Table 1's exact recipe the picture is the same: Total 0.678 against 0.603 for the next-best row, the public scJEPA checkpoint (Appendix Table 10). When its evaluation data is in the pretraining corpus, as here and for most foundation models in Table 2, scRAPTOR is a top-end unsupervised model: only methods fitted on the atlas itself score higher.

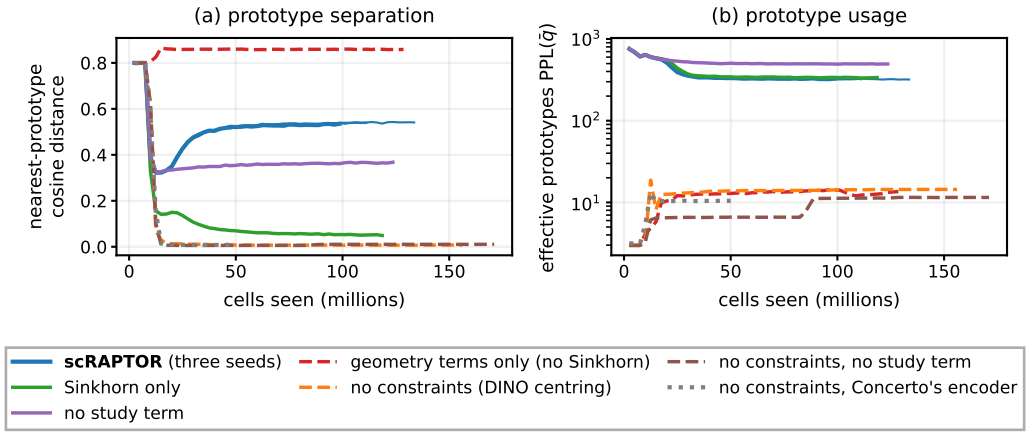

Figure 3: **Without Sinkhorn the prototype head collapses.** Prototype separation (a) and effective number of prototypes (b, log scale) over training for the arms of Table 3. Without Sinkhorn (dashed) usage falls to 12–14 of 1,024 within 20M cells, and to about 10 on Concerto's encoder (dotted; stopped at 50M cells after collapsing); the arms with Sinkhorn keep over 300.

Table 3: **Sinkhorn is load-bearing; the geometry terms are not.** One run per arm at batch 8,192 (our row: mean ± sd of three seeds). Columns: components used; cross-batch accuracy on the corpus test pools; zero-shot Total on the benchmarks and the immune-atlas score of Table 2; linear batch decodability (lower is better); effective prototypes PPL($\bar{q}$). Snapshot: taken during training. Embedding: the 256-d head at each run's selected checkpoint (SIGReg and scVI rows: their own embeddings).

arm	Sink.	geom.	study term	held-out cells	held-out studies	Panc.	PBMC	DKD	immune	batch probe	PPL($\bar{q}$)
scRAPTOR	✓	✓	✓	0.737	0.951	0.614	0.705	0.618	0.579	0.583	335
				±0.001	±0.000	±0.019	±0.009	±0.010	±0.007	±0.030	±6
Sinkhorn only	✓	–	✓	0.741	0.952	0.626	0.715	0.600	0.550	0.606	351
geometry terms only	–	✓	✓	0.643	0.846	0.608	0.539	0.521	0.391	0.283	14
no study term	✓	✓	–	0.739	0.950	0.612	0.682	0.602	0.515	0.688	554
no constraints (DINO centring)	–	–	✓	0.679	0.884	0.580	0.597	0.601	0.446	0.429	14
no constraints, no study term	–	–	–	0.673	0.905	0.590	0.585	0.593	0.431	0.599	12
SIGReg (snapshot)	n/a	n/a	✓	0.662	0.897	0.550	0.661	0.569	0.466	0.776	–
SIGReg	n/a	n/a	–	0.610	0.857	0.529	0.507	0.479	0.448	0.637	6
scVI on our trunk	n/a	n/a	decoder	0.735	0.952	0.573	0.631	0.616	0.481	0.627	–
+ batch adversary	n/a	n/a	decoder	0.732	0.951	0.564	0.628	0.622	–	0.624	–

5.2 Ablations: what is load-bearing

Sinkhorn equipartition is the constraint that matters (Table 3, Figure 3). Without it the head falls back on DINO centring and collapses to 12–14 effective prototypes of 1,024, against 328–554 for every arm with Sinkhorn, and held-out-study accuracy falls to 0.846–0.905 against 0.950–0.952; the immune atlas splits the same way (0.391–0.446 against 0.515–0.584), and every arm without Sinkhorn trails every arm with it on PBMC, by at least 0.084. The tether and KoLeo terms, meant to separate the prototypes, are not needed once Sinkhorn is in place: Sinkhorn alone (351 prototypes) matches or exceeds the full recipe on both corpus pools, Pancreas and PBMC, trailing on DKD (0.600 against 0.618 ± 0.010) and the immune atlas (0.550). Replacing the prototype head by SIGReg (Balestriero & LeCun, 2025), which constrains the trunk output directly, does not help: with the study term the arm is mixed (0.550 / 0.661 / 0.569 at a training snapshot), and without it the trunk collapses, with about 410 of its 512 dimensions exactly constant on every benchmark object by 40M cells and the lowest Totals in the table. Keeping the prototypes in use is therefore what matters, and among the constraints tried only Sinkhorn does it.

The study term lowers batch identity at no in-corpus cost. Without it both corpus pools are unchanged (0.739 / 0.950 against 0.737 ± 0.001 / 0.951 ± 0.000), batch decodability rises (0.688 against 0.583 ± 0.030), and Total falls below the seed range on PBMC and DKD (0.682 / 0.602 against 0.705 ± 0.009 / 0.618 ± 0.010), though not on Pancreas (0.612). Removing it from the unconstrained arm also raises decodability (0.599 against 0.429) and lowers PBMC and DKD (0.585 / 0.593 against 0.597 / 0.601). Low decodability alone is not the aim: the collapsed arms that keep the study term are the least batch-decodable in the table (0.283 and 0.429) and transfer among the worst. The scVI objective on the same trunk is level with scRAPTOR on both corpus pools and more batch-decodable (0.627), and trails it by $+0.041$ on Pancreas and $+0.098$ on the immune atlas. A donor adversary on the trunk, tried on an earlier corpus, costs cross-batch accuracy instead (Appendix B). In these single-run ablations, giving study identity a discardable home helps transfer where removing it adversarially does not.

6 LIMITATIONS

Checkpoint selection is coarse. Across the runs of Table 3 the selection score tracks zero-shot Total ($\rho = 0.90$, Spearman over 12 runs; 0.85 without the two SIGReg arms), so it separates recipes. Within a run it does not: a seed's candidate checkpoints differ by at most 0.002 on the selection mean; the training-time selection metric, computed on the trunk, also stops improving by about 10M cells (Appendix Figure 4). Our held-out studies are easier than the benchmarks: cross-batch accuracy over scored cells is 0.95 there against 0.81–0.89 on the benchmarks (Appendix Table 8). The reported checkpoint is therefore largely a draw among near-equivalent candidates.

Batch metrics penalise real biology. scIB rewards mixing across the batch key, usually the donor, so biological differences between donors score as batch effects: on DKD, rebalancing the donors' cell-type composition raises a principal-component-regression proxy of our batch score from 0.39 ± 0.03 to 0.79 ± 0.01 while scGPT's stays at 0. Part of what the Batch axis penalises on DKD is therefore donor composition, not a technical batch effect.

The ablations are single runs and incomplete. Each arm of Table 3 is one run, so the study-term effect (0.01–0.02 Total on PBMC and DKD, none on Pancreas) lies just outside the seed range and is comparable in size to the spread across seeds. The study term also reduces study identity without removing it: a linear probe still recovers the batch far above chance (0.583 ± 0.030 against 0.09–0.125). Head conditioning on the study and a gradient-reversal adversary were not tested in this framework (Appendix B reports a donor adversary on the trunk on an earlier corpus), and the augmentation was not ablated: no dropout-rate sweep, local-view ablation or count-downsampling comparison. The study-term effect is thus consistent in sign across both pairs of arms, but its size is not yet pinned down.

7 CONCLUSION

An attention-free encoder trained by prototype self-distillation on 86M public cells gives zero-shot embeddings that beat HVG+PCA and the critique's raw-HVG baseline on three benchmarks with no batch label at inference, and lead every other zero-shot embedding we scored on Pancreas, PBMC, the three-dataset mean and the in-corpus immune atlas; DKD does not separate the zero-shot methods. Two components carry this: Sinkhorn equipartition keeps the prototype head from collapsing at scale, and a discardable study term lowers batch identity at no in-corpus cost, with transfer gains on PBMC and DKD.

REPRODUCIBILITY STATEMENT

The supplementary material contains the corpus split (one flag for each of the $112{,}845{,}978$ post-QC cells), the evaluation-pool manifests, per-evaluation metric ledgers, run manifests with resolved arguments, the training and scoring code, the weights of all three seeds, the scored rows behind every table and md5 manifests of every embedding, which the released weights regenerate, so every score can be checked without retraining; the released scripts regenerate every table and figure. The atlas

build package (per-study ingestion configurations and loaders, the $35{,}804$-gene vocabulary, builder, verifier, quality-control lists and an end-to-end smoke test) contains no expression data; accessions are recorded for studies holding 94.6% of training cells. The per-stratum cap and the metadata audits are released as outputs rather than scripts, and the released split file fixes the exact training cells.

ETHICS STATEMENT

This work uses only published, de-identified single-cell RNA-seq data released under their original studies' consent and data-use terms; no new human data were collected. The embeddings are intended for research on cell-type structure and batch effects, not for clinical decisions about individuals.

AI USE STATEMENT

In this work, we used large language model agents (Claude, Anthropic) to draft and revise the manuscript; to implement the training, scoring and analysis code; to orchestrate and monitor training runs; to propose and refine experiments and to help interpret results; and to conduct adversarial reviews whose findings are reflected in the text. We have not used generative AI tools to generate synthetic data sets, and the tasks of formulating mathematical claims and writing proofs are not applicable to this work. The authors chose the claims and approved every experiment run and every result reported. We have reviewed all AI-assisted work: every reported number derives from committed data files and scripts in the supplementary material, and every reference was checked against its published source. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Gabriel Y. Arteaga, Marius Aasan, Rwiddhi Chakraborty, Martine Hjelkrem-Tan, Thalles Silva, Michael Kampffmeyer, and Adín Ramírez Rivera. Why prototypes collapse: Diagnosing and preventing partial collapse in prototypical self-supervised learning. In *ICLR*, 2026. arXiv:2510.20108.

Mojtaba Bahrami, Alejandro Tejada-Lapuerta, Sören Becker, Fatemeh S. Hashemi G., and Fabian J. Theis. scConcept: Contrastive pretraining for technology-agnostic single-cell representations beyond reconstruction. *bioRxiv*, 2025. doi: 10.1101/2025.10.14.682419.

Randall Balestriero and Yann LeCun. LeJEPA: Provable and scalable self-supervised learning without the heuristics. *arXiv preprint arXiv:2511.08544*, 2025.

Dan Busbridge, Jason Ramapuram, Pierre Ablin, Tatiana Likhomanenko, Eeshan Gunesh Dhekane, Xavier Suau, and Russ Webb. How to scale your EMA. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Maren Büttner, Zhichao Miao, F. Alexander Wolf, Sarah A. Teichmann, and Fabian J. Theis. A test metric for assessing single-cell RNA-seq batch correction. *Nature Methods*, 16:43–49, 2019.

Remi Cadene, Corentin Dancette, Hedi Ben-younes, Matthieu Cord, and Devi Parikh. RUBi: Reducing unimodal biases for visual question answering. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *NeurIPS*, volume 33, 2020.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021.

Christopher Clark, Mark Yatskar, and Luke Zettlemoyer. Don't take the easy way out: Ensemble based methods for avoiding known dataset biases. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 4069–4082, 2019.

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*, 21:1470–1480, 2024.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in Neural Information Processing Systems*, volume 26, 2013.

Juan Javier Díaz-Mejía, Elias Williams, Octavian Focsa, Dylan Mendonca, Swechha Singh, Brendan Innes, and Sam Cooper. Benchmarking and optimizing organism wide single-cell RNA alignment methods. In *ICLR 2025 Workshop on Learning Meaningful Representations of Life (LMRL)*, 2025. arXiv:2503.20730.

Jiarui Ding, Xian Adiconis, Sean K. Simmons, Monika S. Kowalczyk, Cynthia C. Hession, Nemanja D. Marjanovic, Travis K. Hughes, Marc H. Wadsworth, Tyler Burks, Lan T. Nguyen, et al. Systematic comparison of single-cell and single-nucleus RNA-sequencing methods. *Nature Biotechnology*, 38(6):737–746, 2020. doi: 10.1038/s41587-020-0465-8.

C. Domínguez Conde, C. Xu, L. B. Jarvis, et al. Cross-tissue immune cell analysis reveals tissue-specific features in humans. *Science*, 376(6594):eabl5197, 2022.

A. ElSheikh, R.-X. Wang, W. Wu, Y. Wen, P. Dibaeinia, J. Y. Zhang, J. Y.-C. Hu, M. Knudson, S. Babu, S.-H. Sun, A. A. Khan, and H. Liu. Cell-JEPA: Latent representation learning for single-cell transcriptomics. *arXiv preprint arXiv:2602.02093*, 2026.

Hariprasath Govindarajan, Per Sidén, Jacob Roll, and Fredrik Lindsten. On partial prototype collapse in the DINO family of self-supervised methods. In *BMVC*, 2024. arXiv:2410.14060.

He He, Sheng Zha, and Haohan Wang. Unlearn dataset bias in natural language inference by fitting the residual. In *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo)*, pp. 132–142, 2019.

Graham Heimberg, Tony Kuo, Daryle J. DePianto, et al. A cell atlas foundation model for scalable search of similar human cells. *Nature*, 638:1085–1094, 2025.

Elad Hoffer, Itay Hubara, and Daniel Soudry. Fix your classifier: the marginal value of training the last weight layer. In *ICLR*, 2018.

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.

Ilya Korsunsky, Nghia Millard, Jean Fan, Kamil Slowikowski, Fan Zhang, Kevin Wei, Yuriy Baglaenko, Michael Brenner, Po-ru Loh, and Soumya Raychaudhuri. Fast, sensitive and accurate integration of single-cell data with Harmony. *Nature Methods*, 16:1289–1296, 2019.

Xiangjie Li, Kui Wang, Yafei Lyu, Huize Pan, Jingxiao Zhang, Dwight Stambolian, Katalin Susztak, Muredach P. Reilly, Gang Hu, and Mingyao Li. Deep learning enables accurate clustering with batch effect removal in single-cell RNA-seq analysis. *Nature Communications*, 11:2338, 2020. doi: 10.1038/s41467-020-15851-3.

E. Litman, T. Myers, V. Agarwal, E. Mittal, O. Li, A. Gopinath, and T. Kassis. GeneJEPA: A predictive world model of the transcriptome. *bioRxiv*, 2025. 2025.10.14.682378.

Romain Lopez, Jeffrey Regier, Michael B. Cole, Michael I. Jordan, and Nir Yosef. Deep generative modeling for single-cell transcriptomics. *Nature Methods*, 15:1053–1058, 2018.

Malte D. Luecken, Maren Büttner, Kridsadakorn Chaichoompu, et al. Benchmarking atlas-level data integration in single-cell genomics. *Nature Methods*, 19:41–50, 2022.

Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. Long-tail learning via logit adjustment. In *ICLR*, 2021.

Kevin P. Nguyen, Alex H. Treacher, and Albert A. Montillo. Adversarially-regularized mixed effects deep learning (ARMED) models improve interpretability, performance, and generalization on clustered (non-iid) data. *IEEE TPAMI*, 45(7):8081–8093, 2023.

Open Problems Consortium. Open problems in single-cell analysis: Batch integration and label projection benchmarks (results v2.0.0; CellPLM row from v2.1.0-rc2). `https://openproblems.bio/benchmarks/foundation_models`, 2026. Accessed 22 September 2026.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, et al. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. arXiv:2304.07193.

Olga Ovcharenko, Florian Barkmann, Philip Toma, Imant Daunhawer, Julia Vogt, Sebastian Schelter, and Valentina Boeva. scSSL-Bench: Benchmarking self-supervised learning for single-cell data. In *ICML*, 2025. arXiv:2506.10031.

Yanay Rosen, Yusuf Roohani, Ayush Agrawal, Leon Samotorčan, Tabula Sapiens Consortium, Stephen R. Quake, and Jure Leskovec. Universal cell embedding provides a foundation model for cell biology. *Nature*, 656(8126):183–191, 2026. doi: 10.1038/s41586-026-10689-z.

Alexandre Sablayrolles, Matthijs Douze, Cordelia Schmid, and Hervé Jégou. Spreading vectors for similarity search. In *International Conference on Learning Representations*, 2019.

Valentine Svensson. Droplet scRNA-seq is not zero-inflated. *Nature Biotechnology*, 38(2):147–150, 2020. doi: 10.1038/s41587-019-0379-5.

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, and Patrick T. Ellinor. Transfer learning enables predictions in network biology. *Nature*, 618:616–624, 2023.

Yao-Hung Hubert Tsai, Tianqin Li, Martin Q. Ma, Han Zhao, Kun Zhang, Louis-Philippe Morency, and Ruslan Salakhutdinov. Conditional contrastive learning with kernel. In *International Conference on Learning Representations*, 2022.

Yuhao Wang, Zelin Zang, Yuxuan Liu, Zhen Lei, and Stan Z. Li. BioM-JEPA: Joint-embedding prediction of graph-connected gene blocks in single cells. *arXiv preprint arXiv:2608.05928*, 2026.

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.

Parker C. Wilson, Haojia Wu, Yuhei Kirita, Kohei Uchimura, Nicolas Ledru, Helmut G. Rennke, Paul A. Welling, Sushrut S. Waikar, and Benjamin D. Humphreys. The single-cell transcriptomic landscape of early human diabetic nephropathy. *Proceedings of the National Academy of Sciences*, 116(39):19619–19625, 2019. doi: 10.1073/pnas.1908706116.

Parker C. Wilson, Yoshiharu Muto, Haojia Wu, Anil Karihaloo, Sushrut S. Waikar, and Benjamin D. Humphreys. Multimodal single cell sequencing implicates chromatin accessibility and genetic background in diabetic kidney disease progression. *Nature Communications*, 13:5253, 2022. doi: 10.1038/s41467-022-32972-z.

Chenling Xu, Romain Lopez, Edouard Mehlman, Jeffrey Regier, Michael I. Jordan, and Nir Yosef. Probabilistic harmonization and annotation of single-cell transcriptomics data with deep generative models. *Molecular Systems Biology*, 17(1):e9620, 2021.

Meng Yang, Yueyuxiao Yang, Chenxi Xie, Ming Ni, Jian Liu, Huanming Yang, Feng Mu, and Jian Wang. Contrastive learning enables rapid mapping to multimodal single-cell atlas of multimillion scale. *Nature Machine Intelligence*, 4(8):696–709, 2022.

Yang You, Jing Li, Sashank Reddi, Jonathan Hseu, Sanjiv Kumar, Srinadh Bhojanapalli, Xiaodan Song, James Demmel, Kurt Keutzer, and Cho-Jui Hsieh. Large batch optimization for deep learning: Training BERT in 76 minutes. In *International Conference on Learning Representations*, 2020.

Nicholas D. Youngblut, Christopher Carpenter, Arshia Nayebnazar, Abhinav Adduri, Rohan Shah, Chiara Ricci-Tam, Jaanak Prashar, Rajesh Ilango, Noam Teyssier, Silvana Konermann, Patrick D. Hsu, Alexander Dobin, Dave P. Burke, Hani Goodarzi, and Yusuf H. Roohani. scBaseCount: An AI agent-curated, standardized, auto-updated single-cell data repository. *Cell*, 189(19):5932–5944.e6, 2026. doi: 10.1016/j.cell.2026.08.025.

Yuyao Zhai, Weixu Wang, Till Richter, and Fabian J. Theis. scJEPA: Dual-space self-supervised learning for single-cell representations. In *LMRL Workshop at ICLR*, 2026.

Jesse Zhang, Airol A. Ubas, Richard de Borja, Valentine Svensson, Nicole Thomas, Neha Thakar, Ian Lai, Aidan Winters, Umair Khan, Matthew G. Jones, et al. Tahoe-100M: A giga-scale single-cell perturbation atlas for context-dependent gene function and cellular modeling. *bioRxiv*, 2025. doi: 10.1101/2025.02.20.639398.

A METHOD, CORPUS AND EXPOSURE DETAILS

Table 4: **Every setting of the reported recipe.** Values are those of §3, §4 and this appendix.

Data	
corpus	CorpusX v2.0.0, training split: 86,112,911 cells, 1,065 studies, 25,029 samples (of 112,845,978 post-QC cells, 1,077 studies)
quality control, cap	≥ 300 detected genes; $\leq 100{,}000$ cells per (study, tissue, disease) stratum, donor-balanced round-robin
input	raw counts over 35,804 genes; $\log(1 + 10^4 x / \sum_g x_g)$ over the full vocabulary, then gather to 3,000 HVGs (normalised dispersion on the training cells)
Model	
trunk	gene table $3{,}000 \times 512$, $\mathcal{N}(0, 0.02^2)$ init (1,536,000); linear $512 \to 512$ with bias, LayerNorm, ReLU, Dropout 0.1 (263,680); no attention
embedding	ℓ_2-normalised 256-d head bottleneck of the teacher; 7.57M parameters at inference (trunk 1.80M, MLP 5.77M)
head	MLP $512 \to 2048 \to 2048 \to 256$, GELU, trunc-normal init 0.02; ℓ_2 normalisation; weight-normalised $256 \to 1{,}024$ prototypes, no bias, gain fixed at 1 (MLP 5.77M, used at inference; prototypes 0.26M, training only)
assignment	Sinkhorn–Knopp, uniform marginal, $\varepsilon = 0.05$, 3 iterations, queue 4,096 teacher logits; student temperature 0.1; teacher temperature 0.04 (does not reach the loss under Sinkhorn)
geometry terms	tether $\lambda = 0.1$ to a fixed frame with entries $\pm 1/16$; KoLeo $\lambda = 0.05$ on prototypes
study term	rank 128 over 1,077 study identities, zero-initialised projection, added to student and teacher logits (0.27M parameters); discarded at inference
Views	
gene dropout	per cell and view, on raw counts: rate $\sim U[0.2, 0.5]$ (global), $U[0.5, 0.8]$ (local)
views per cell	teacher 2 global; student 2 global +8 local; 18 loss pairs
Optimisation	
optimiser	LAMB, $\beta = (0.9, 0.999)$, $\epsilon = 10^{-6}$, trust ratio clipped to $[0, 10]$; weight decay 0.04 on every parameter; gradient clipping 3.0
learning rate	10^{-4} at reference batch 128, scaled linearly to 6.4×10^{-3}; linear warm-up 2,000 batches (16.4M cells)
batch, teacher	global batch 8,192; EMA momentum 0.996 at batch 128, rescaled to $0.996^{64} = 0.7737$; prototypes frozen for 1,000 batches (8.2M cells)
precision	gene-table product in `float16` with `float32` accumulation; everything else `float32`
length	early stop after 20 evaluations without improvement on either pool: seeds 20 / 21 / 22 stopped at 99M / 134M / 111M cells (39 / 53 / 44 evaluations); continuations to the two-epoch cap of 172M cells
Selection	
pools	held-out cells: 250,000 cells of the training studies (batch = study, 72 types, 96,326 labelled); held-out studies: 355,654 cells of four studies (batch = sample, 25 types, 292,823 labelled); test halves 250,000 and 350,894 cells
metric	$\mathrm{acc}_{\mathrm{cross}}$, $k = 50$, median-centred and IQR-scaled embeddings, every 2.5M cells
rule	best mean of the two pools' $\mathrm{acc}_{\mathrm{cross}}$ at the 256-d head among the ledger's five best evaluations, rescored exactly; warm-up evaluations not eligible; selected at 35M / 67M / 50M cells (steps 4,270 / 8,235 / 6,100)
Compute and scoring	
compute	one A10G GPU per seed; 13.8 / 21.1 / 15.4 h to early stop for seeds 20 / 21 / 22 ($\approx$ 2,000 cells/s including evaluation)
scoring	scIB metrics 0.5.1 on 4,000 batch-aware HVGs, raw and unscaled, Total $= 0.6\,\mathrm{Bio} + 0.4\,\mathrm{Batch}$; four to eight passes per headline cell; GPU and CPU backends agree on every non-Leiden metric to 2.3×10^{-4}

Trunk and head. Gene table $3{,}000 \times 512$, $\mathcal{N}(0, 0.02^2)$ init; one linear $512 \to 512$ with bias, LayerNorm, ReLU, Dropout 0.1. No attention module is instantiated; the forward pass is one dense matrix product, computed in `float16` with `float32` accumulation and exact to 5×10^{-7} against the explicit per-gene sum. Head: trunc-normal init 0.02; weight-normalised $256 \to 1{,}024$ without bias, gain fixed at 1. The teacher temperature does not reach the loss under Sinkhorn. Sinkhorn: uniform target, queue updated only in training, all-gathered across ranks so every rank solves the same problem. Tether: $\lambda_a \, \mathbb{E}_k \|\hat{w}_k - a_k\|^2$ with $\lambda_a = 0.1$, frame from a fixed seed, entries $\pm 1/16$.

KoLeo: $-\lambda_k \mathbb{E}_k \log \|\hat{w}_k - \hat{w}_{\mathrm{nn}(k)}\|$ with $\lambda_k = 0.05$ on prototypes. Study term: rank 128, 0.27M parameters, embeddings $\mathcal{N}(0, 0.02^2)$, projection zero-initialised, excluded at inference. At inference the gene table, the linear layer and the head MLP remain, 7.57M parameters in all.

Optimisation. LAMB, $\beta = (0.9, 0.999)$, $\epsilon = 10^{-6}$, trust ratio clipped to $[0, 10]$, no no-decay group. The teacher-momentum ramp toward 1 leaves it effectively constant over the run. During the 1,000 frozen-head batches the tether and KoLeo gradients on the prototypes are also discarded. Only the gene-table product runs in `float16`. No setting differs between the three seeds except the training seed.

Evaluation. $k = 50$ nearest neighbours on median-centred, IQR-scaled embeddings, every 2.5M cells. Held-out cells: 250,000 cells of the training studies, batch axis study, 96,326 of them labelled and scored, exact search. Held-out studies: 355,654 cells of the four held-out studies, batch axis sample, 292,823 labelled; the training-time scorer uses an approximate nearest-neighbour search on this pool, which reads about 0.01 below the exact value. The three seeds stopped at 99M, 134M and 111M cells (39, 53 and 44 evaluations). Because of the approximate search, the reported checkpoint is always rescored exactly.

Split. CorpusX v2.0.0 holds 112,845,978 post-QC cells (minimum 300 detected genes) from 1,077 studies. The training manifest carries one flag per cell: the training split, the two in-corpus pools (250,000 random training-study cells each, removed from training), and the two out-of-corpus pools (355,654 and 350,894 cells: the four wholly held-out studies split by sample into two halves, so no sample straddles them). Training cells are capped at 100,000 per (study, tissue, disease) stratum by donor-balanced round-robin sampling. Eight studies (511,201 cells) are flagged as log-transformed rather than raw counts and are excluded from every split. The HVGs are the top 3,000 genes by normalised dispersion, computed in gene-blocks across the training cells, so the panel is chosen on training cells only, never on an evaluation set.

Benchmark exposure. Pancreas and PBMC (scSSL-Bench) are absent from the corpus. Open Problems DKD (Wilson et al., 2022) reuses an accession of Wilson et al. (2019), which is in the corpus; barcode overlap is 41.3% against 2.7% expected by chance. All 16,155 corpus cells with a barcode match to the DKD object are removed from every split; the accession's remaining cells stay in training, which is the exposure we disclose. A further 1,921 cells overlapping an external benchmark are likewise removed. The immune atlas is in the corpus and is reported only as an in-corpus test (§5.1); of the three benchmarks, only DKD shares an accession with the corpus.

B A BATCH ADVERSARY ON THE TRUNK

A batch adversary on the trunk removes cell-type structure along with batch identity. In an earlier 20M-cell campaign on a previous corpus version (same objective and global batch, one seed), a discriminator trained on a detached copy of the 512-d trunk output predicted the donor within each (study, tissue, disease) stratum, and the encoder received its confusion loss at weight 0.05. Against the matched control (Table 5), the adversary loses cross-batch accuracy at every evaluation but one, ending at 0.420 against 0.692 on held-out cells and 0.729 against 0.870 on held-out studies at 20M cells. Zero-shot it scores 0.577 / 0.647 / 0.557 against the control's 0.606 / 0.640 / 0.598, and on the Open Problems DKD task it lowers the batch-integration score (0.475 against 0.549) and label-projection accuracy (0.864 against 0.927), halving ARI (0.33 against 0.65). Adversarial batch terms help reconstruction models such as BA-scVI (Díaz-Mejía et al., 2025), but in self-distillation this one trades away the structure the embedding is for, which is why the study term takes the discardable route instead.

Table 5: **A batch adversary on the trunk costs cross-batch accuracy.** Training-time cross-batch accuracy on held-out cells (in) and studies (out) at matched cells seen; earlier campaign, single runs.

	5M cells		10M cells		15M cells		20M cells	
arm	in	out	in	out	in	out	in	out
control (no adversary)	0.655	0.865	0.666	0.783	0.678	0.866	0.692	0.870
donor adversary on the trunk	0.595	0.800	0.545	0.855	0.567	0.814	0.420	0.729

C WHICH METRICS CARRY THE MARGINS

This appendix asks how much of Table 1's margins depend on a single metric and on batch mixing. Table 7 recomputes Total with each of the ten scIB metrics removed in turn: on PBMC scRAPTOR stays ahead of HVG+PCA whichever metric is dropped (smallest gap $+0.173$, without PCR), whereas on Pancreas and DKD the gap disappears without PCR (-0.003 and -0.003). Table 8 splits strict cross kNN into accuracy over the cells that have an other-batch neighbour and the share of such cells: HVG+PCA transfers labels well where its batches touch but has an other-batch neighbour for only 45% of Pancreas cells. Together, the two tables show that the PBMC margin is broad-based and that the Pancreas and DKD margins rest on PCR.

Table 6: Per-metric raw scIB values behind Table 1. Each Bio metric contributes 0.12 and each Batch metric 0.08 to Total. iso: isolated labels; ASW_l / ASW_b: silhouette label / batch; GC: graph connectivity.

dataset	embedding	iso	NMI	ARI	ASW_l	cLISI	ASW_b	iLISI	kBET	GC	PCR
Pancreas	scRAPTOR (three-seed mean)	0.611	0.639	0.532	0.541	0.994	0.762	0.157	0.360	0.775	0.647
	scVI on our trunk	0.611	0.624	0.457	0.548	0.997	0.890	0.060	0.280	0.815	0.259
	scGPT (scRAPTOR's panel)	0.637	0.530	0.356	0.514	0.993	0.887	0.036	0.200	0.756	0.000
	HVG+PCA	0.743	0.690	0.423	0.591	1.000	0.824	0.001	0.234	0.864	0.046
PBMC	scRAPTOR (three-seed mean)	0.870	0.696	0.588	0.660	0.996	0.789	0.289	0.294	0.901	0.823
	scVI on our trunk	0.630	0.735	0.671	0.563	0.996	0.874	0.158	0.136	0.769	0.554
	scGPT (scRAPTOR's panel)	0.576	0.658	0.491	0.538	0.987	0.898	0.146	0.145	0.872	0.471
	HVG+PCA	0.625	0.504	0.164	0.534	0.998	0.862	0.001	0.022	0.837	0.070
DKD	scRAPTOR (three-seed mean)	0.604	0.674	0.478	0.648	0.999	0.730	0.241	0.314	0.861	0.475
	scVI on our trunk	0.579	0.754	0.685	0.609	1.000	0.883	0.145	0.157	0.728	0.351
	scGPT (scRAPTOR's panel)	0.588	0.782	0.733	0.628	1.000	0.888	0.161	0.181	0.861	0.289
	HVG+PCA	0.631	0.769	0.570	0.626	1.000	0.865	0.023	0.166	0.893	0.134

Table 7: **Drop-one-metric analysis of Table 1.** Total of our mean and HVG+PCA, and their gap, with each metric removed; on Pancreas and DKD the gap rests on PCR.

	Pancreas			PBMC			DKD		
metric dropped	ours	PCA	gap	ours	PCA	gap	ours	PCA	gap
none (Table 1)	0.614	0.571	$+0.043$	0.705	0.482	$+0.223$	0.618	0.598	$+0.020$
isolated labels	0.622	0.563	$+0.059$	0.689	0.473	$+0.215$	0.629	0.611	$+0.018$
NMI	0.618	0.571	$+0.047$	0.715	0.491	$+0.223$	0.619	0.591	$+0.028$
ARI	0.634	0.611	$+0.023$	0.731	0.542	$+0.189$	0.648	0.620	$+0.028$
ASW label	0.632	0.586	$+0.046$	0.720	0.487	$+0.233$	0.623	0.612	$+0.011$
cLISI	0.564	0.525	$+0.040$	0.670	0.417	$+0.252$	0.570	0.556	$+0.014$
ASW batch	0.592	0.528	$+0.064$	0.688	0.432	$+0.256$	0.597	0.553	$+0.044$
iLISI	0.652	0.611	$+0.042$	0.738	0.518	$+0.220$	0.646	0.637	$+0.009$
kBET	0.632	0.587	$+0.045$	0.737	0.516	$+0.221$	0.639	0.623	$+0.016$
graph conn.	0.591	0.524	$+0.066$	0.677	0.434	$+0.242$	0.584	0.550	$+0.034$
PCR	0.603	0.606	-0.003	0.685	0.511	$+0.173$	0.623	0.626	-0.003

Table 8: **Cross-batch kNN label transfer.** Strict: accuracy of the vote among the other-batch cells in each cell's 50 nearest neighbours, counting a cell with none as wrong (Table 1); scored only: the same over cells with at least one; scored: their share. Strict = scored only × scored; raw Euclidean distances. Our row: three-seed mean (sd below).

	Pancreas			PBMC			DKD		
embedding	strict	scored only	scored	strict	scored only	scored	strict	scored only	scored
scRAPTOR	0.790 ±0.023	0.808 ±0.010	98% ±2%	0.886 ±0.003	0.886 ±0.003	100% ±0%	0.866 ±0.006	0.880 ±0.008	98% ±1%
scVI (our corpus)	0.674	0.805	84%	0.855	0.868	99%	0.891	0.912	98%
scVI on our trunk	0.718	0.826	87%	0.878	0.881	100%	0.822	0.920	89%
scJEPA (public checkpoint)	0.622	0.688	90%	0.807	0.868	93%	0.824	0.840	98%
HVG+PCA	0.434	0.957	45%	0.413	0.865	48%	0.654	0.942	69%
corpus PCA-512 + LN + ReLU	0.600	0.889	68%	0.814	0.868	94%	0.781	0.932	84%
corpus PCA-512, mean-centred	0.611	0.889	69%	0.793	0.828	96%	0.795	0.929	86%
random projection-512 + LN + ReLU	0.678	0.933	73%	0.827	0.856	97%	0.865	0.925	93%

D SEEDS, SELECTION AND SCORING NOISE

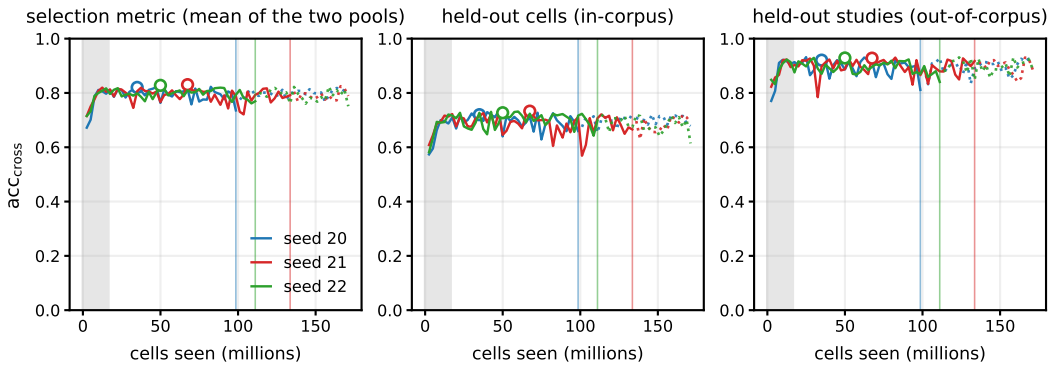

Figure 4: **The selection metric plateaus early.** Ledger validation trace of the three seeds (training-time, computed on the trunk): the selection metric and its two pools. Grey: warm-up; solid: to early stop; dotted: continuation to the cap; circles: reported checkpoints, chosen at the head (§4).

Untrained controls. Table 1's controls ask what the architecture and a fixed projection give without our objective: our encoder at its training initialisation (trunk, head MLP and ℓ_2, with seed 20's constructor and no training); a corpus PCA to 512 dimensions fitted on 2M training cells, uncentred, followed by the trunk's LayerNorm and ReLU; a Gaussian random projection of the same shape; and the top 256 axes of that PCA, centred and ℓ_2-normalised. All read the same 3,000-gene input as scRAPTOR.

Every model row in Table 1 is a deterministic embedding of one checkpoint per training seed; the scRAPTOR row is the mean over three seeds and their scoring passes.

Scoring. Re-scoring one embedding changes raw scIB Total only through the Leiden clustering: every non-Leiden metric reproduces to six decimals, and Total spans about 0.009 (Pancreas), 0.011 (DKD) and 0.002 (PBMC) across passes, so rows are means of four to eight passes (sd of Total at most 0.006). GPU and CPU backends agree on every non-Leiden metric to 2.3×10^{-4}, and kBET's unseeded subsampling adds an sd of 2–5 $\times 10^{-4}$. The immune label-projection test set is a single held-out batch. Scoring noise is therefore small beside the spread across seeds.

Comparators. Concerto's training-time selection metric is invalid on its sparse ReLU output, so every post-warm-up Concerto checkpoint was rescored exactly. Both scGPT rows of Table 1 run the Open Problems scGPT component with scGPT's whole-human weights and its published call

(up to 1,200 expressed genes per cell), one on exactly scRAPTOR's gene panel (2,651 / 2,840 / 2,785 of its genes present in Pancreas / PBMC / DKD) and one, transductive, on the tutorial's 3,000 seurat_v3 HVGs selected on each evaluation set. Our reproduction of the scJEPA objective uses the released inference code's architecture, since the training code is not public: an MLP encoder 35,804–512–512–256–256–64 with LayerNorm on the full vocabulary, a 64-d projection head and a mirrored decoder, 37.7M parameters in all. We trained it on our corpus at batch 8,192 with one global and eight masked views per cell (nine encoder passes per cell per step) and the N-scaled SIGReg statistic of the original, to the same two-epoch cap as our seeds (172M cells, 16.0 h on one A10G at 2,985 cells/s including evaluation), and report the final checkpoint at the cap. Table 9 lists size, cells seen and training compute for the models we trained or embedded (Concerto's run is described above): the comparators we trained all use our corpus, and published ones are used as released.

Table 9: Size, cells seen and training compute (one A10G) of the Table 1 models; "published": used as released.

model	params	dim	cells seen	training compute (one A10G)	batch label at inference
scRAPTOR (each seed)	7.6M	256	99–134M (reported: 35–67M)	13.8–21.1 h to early stop ($\approx$ 2,000 cells/s incl. evaluation)	no
scVI (our corpus)	2.1M	30	97M	8.1 h to early stop (3,378 cells/s)	no (decoder only)
scVI on our trunk	3.1M	30	67M	6.5 h to early stop (2,864 cells/s)	no (decoder only)
scGPT (whole-human)	51M	512	33M	published	no
Geneformer (12-layer)	–	512	95M	published	no
scJEPA (public checkpoint)	–	256	33M	published	no
scJEPA (our reproduction, lr 10^{-3})	37.7M	64	172M	16.0 h to the two-epoch cap (2,985 cells/s incl. evaluation; 9 encoder passes per cell)	no
HVG+PCA	–	50	–	fitted on each evaluation set	yes (batch-aware HVGs)

Table 10: **The immune atlas under Table 1's recipe.** Full object (329,762 cells); in-corpus for every row but HVG+PCA; cross kNN as in Table 1. Our row: three-seed mean (sd below); bold: best per column.

embedding	Bio	Batch	Total	cross kNN	passes
scRAPTOR	**0.741** $\pm$0.010	**0.583** $\pm$0.010	**0.678** $\pm$0.004	**0.857**	12
scJEPA (public checkpoint)	0.639	0.548	0.603	0.744	4
scVI (our corpus)	0.656	0.477	0.584	0.702	4
scJEPA (our reproduction)	0.631	0.478	0.570	0.554	4
scVI on our trunk	0.650	0.444	0.568	0.682	4
scGPT	0.627	0.472	0.565	0.627	4
HVG+PCA	0.638	0.454	0.564	0.493	4