HI-CELL: A CELL-LEVEL FOUNDATION MODEL FOR SINGLE-CELL HI-C DATA

Anonymous authors
Paper under double-blind review

ABSTRACT

Single-cell Hi-C (scHi-C) technology provides a direct view of three-dimensional genome organization in individual cells, but its high dimensionality, extreme sparsity, and multiscale contact patterns create an urgent need for a sophisticated and versatile computational framework. Here we introduce HI-CELL, a cell-level foundation model pretrained on Human-Hi-C-Corpus, our curated large-scale collection of 324,102 single cells from 17 human tissues and 357 bulk Hi-C experiments. HI-CELL represents each cell as a hierarchical "cell sentence" composed of cell-, chromosome-, and multiscale patch-level tokens, and uses global bidirectional self-attention to integrate local contact patterns with chromosome-wide and whole-cell organization. A three-stage progressive pretraining strategy combining single-cell pretraining, bulk-derived pseudo-single-cell supervision, and cross-depth refinement enables HI-CELL to learn transferable cellular representations and structural priors. Across six unseen scHi-C datasets, zero-shot embeddings are competitive with dedicated methods, while label-free fine-tuning achieves the best mean clustering performance among the evaluated baselines. Frozen HI-CELL representations also support accurate cell-type annotation, achieving the highest mean macro-F1 among baselines. Beyond cell representation, HI-CELL performs zero-shot contact imputation across sequencing depths, leading all evaluated methods in held-out and leave-one-out structural correlation on all six datasets and preserving independently validated A/B compartment profiles. Together, HI-CELL provides a unified and transferable framework for analyzing cellular identity and reconstructing multiscale 3D genome organization from sparse scHi-C data.

1 INTRODUCTION

The three-dimensional genome provides a structural view of cellular identity, as cells with the same DNA sequence can organize their chromosomes differently as they specialize and change state. By measuring chromatin contacts in individual cells, single-cell Hi-C (scHi-C) enables direct investigation of this structural heterogeneity and its relationship to cell type and state (Ramani et al., 2017; Lee et al., 2019). However, scHi-C data are exceptionally difficult to analyze. Each cell captures only a small fraction of all possible contacts, sequencing depth varies widely, and contacts are distributed unevenly across genomic distances, producing strong near-diagonal signals. Meanwhile, informative chromatin patterns extend from local interaction neighborhoods to chromosome-wide organization. This combination of high dimensionality, extreme sparsity, and multiscale contact patterns makes it challenging to obtain reliable cell representations and recover the underlying chromatin structure.

Existing methods address these challenges through contact smoothing, topic modeling, graph-based representation learning, and multiscale modeling (Zhou et al., 2019; Kim et al., 2020; Zhang et al., 2022; Zhou & Wu, 2025; Wu et al., 2026). These methods have substantially advanced scHi-C clustering, imputation, and annotation, but most are trained or fitted separately on each target dataset. Consequently, every new dataset must provide enough information both to learn basic structural regularities and to distinguish its constituent cells. This requirement is particularly restrictive for sparse scHi-C data and leaves potentially transferable knowledge shared across tissues, experimental protocols, and biological states largely unused.

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.

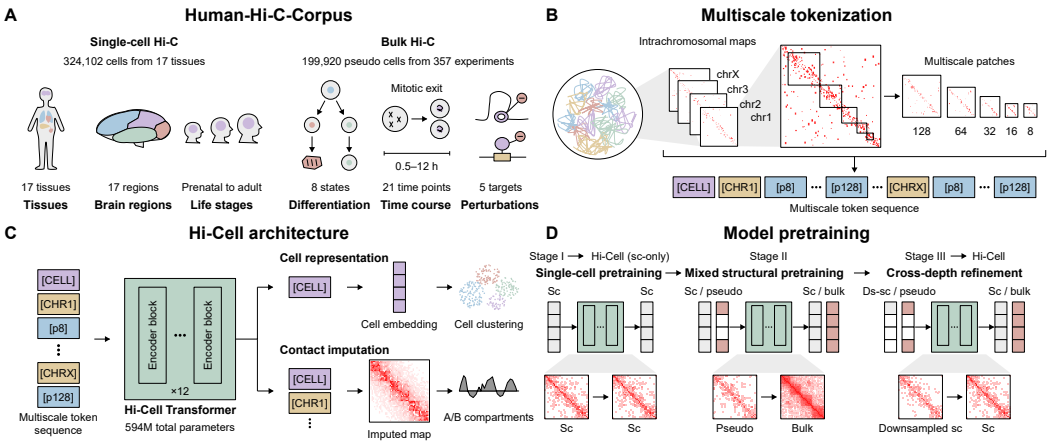


Figure 1: **Overview of Hi-Cell. A**, Human-Hi-C-Corpus; totals include training and validation data. **B**, multiscale tokenization. CELL, CHR, and p denote cell, chromosome, and patch tokens; patch numbers give side lengths in bins. **C**, cell representation and contact imputation with the shared Transformer. **D**, three-stage progressive pretraining and input–target pairs. Sc, single-cell Hi-C; Ds-sc, downsampled single-cell Hi-C; pseudo, pseudo-single cells; bulk, bulk Hi-C; Hi-Cell (sc-only), the Stage I checkpoint.

Pretraining offers a way to learn such shared knowledge from large data collections and reuse it in new settings. Foundation models for single-cell transcriptomics and epigenomics have demonstrated the value of this strategy across diverse downstream tasks (Cui et al., 2024; Hao et al., 2024; Chen et al., 2025a). Their gene- or regulatory-element-based representations, however, do not naturally encode pairwise chromatin contacts or their dependence on genomic scale. HiCFoundation extends pretraining to Hi-C contact submatrices and supports a broad range of structural analyses, including single-cell applications (Wang et al., 2026). A complementary direction is to organize pretraining around the cell itself, jointly learning cellular identity and chromatin structure from complete single-cell contact profiles.

Here we introduce HI-CELL (Figure 1), a cell-level foundation model pretrained on Human-Hi-C-Corpus, our curated collection of 324,102 single cells from 17 human tissues and 357 bulk Hi-C experiments. HI-CELL represents each cell as a hierarchical "cell sentence" composed of cell-, chromosome-, and multiscale patch-level tokens, and uses global bidirectional self-attention to integrate local contact patterns with chromosome-wide and whole-cell organization. A three-stage progressive pretraining strategy combining single-cell pretraining, bulk-derived pseudo-single-cell supervision, and cross-depth refinement enables the model to learn transferable cellular representations and structural priors. Across six unseen scHi-C datasets, zero-shot embeddings are competitive with dedicated methods, while label-free fine-tuning achieves the best mean clustering performance among the evaluated methods. Frozen representations also achieve the highest mean macro-F1 among the evaluated annotation approaches. Without fine-tuning, the same pretrained model leads all evaluated methods in held-out and leave-one-out structural correlation on every dataset and preserves independently validated A/B compartment profiles. Together, these results establish HI-CELL as a unified framework for analyzing cellular identity and reconstructing multiscale 3D genome organization from sparse scHi-C data.

2 RELATED WORK

2.1 COMPUTATIONAL MODELS FOR SINGLE-CELL HI-C

The three-dimensional (3D) organization of chromatin contributes to gene regulation by shaping spatial interactions between genes and regulatory elements (Wu et al., 2026). Single-cell Hi-C (scHi-C) measures pairwise chromatin contacts within individual cells, enabling the study of genome organization across cell types and states (Ramani et al., 2017; Lee et al., 2019). Its sparse observations,

strong dependence on genomic distance, and multiscale contact patterns create distinct challenges for computational analysis.

Existing methods address these challenges through representation learning, contact imputation, and cell-type annotation. Topic modeling extracts cell-type-associated compartment patterns (Kim et al., 2020), while scHiCluster combines convolution and random-walk smoothing for imputation and cell representation (Zhou et al., 2019). Deep learning methods include Higashi, which jointly learns cell embeddings and contact predictions through hypergraph modeling (Zhang et al., 2022); scVI-3D and Va3DE, which use variational models for denoising and embedding (Zheng et al., 2022; Plummer et al., 2025); and Hi-Cformer, which integrates chromatin contact patterns across genomic scales (Wu et al., 2026). For supervised annotation, scHiClassifier combines multiple contact-derived feature sets to predict cell types (Zhou & Wu, 2025).

Despite supporting diverse downstream analyses, most of these approaches learn from individual datasets or task-specific training collections rather than a shared, broadly pretrained model. They therefore rely on the available training dataset to estimate structural regularities, even when contact coverage is limited, and often require additional fitting for new datasets. The growing availability of scHi-C collections makes an alternative possible: learning recurring contact patterns across studies and transferring them to new cellular analyses. This motivates a foundation model that supplies reusable representations and reconstruction capabilities from a common pretrained backbone.

HiCFoundation introduces large-scale pretraining on Hi-C contact submatrices and supports structural, multiomics, and single-cell analyses (Wang et al., 2026). Its pretraining operates on genomic regions, whose embeddings can also be aggregated into broader genomic summaries. HI-CELL instead explicitly learns a whole-cell representation jointly with patch- and chromosome-level structure. The distinction lies in organizing pretraining around cellular representation and contact reconstruction together, providing a shared interface for cell-level tasks and zero-shot imputation.

2.2 FOUNDATION MODELS FOR SINGLE-CELL OMICS

Geneformer, scGPT, and scFoundation demonstrate the value of pretraining on transcriptomic collections for downstream cellular analysis (Theodoris et al., 2023; Cui et al., 2024; Hao et al., 2024). EpiAgent extends this approach to chromatin accessibility, supporting feature extraction, cell-type annotation, and imputation (Chen et al., 2025a). However, their input vocabularies and learning objectives are organized around genes or regulatory elements, whereas each scHi-C observation links two genomic loci. The interpretation of a contact further depends on genomic separation and its surrounding chromosomal structure. Consequently, transferring these models to scHi-C requires adapting both the input representation and the learning objective, rather than directly substituting contact counts for expression or accessibility values. HI-CELL addresses this modality-specific requirement through contact-aware pretraining that integrates local structure with chromosome and whole-cell context.

3 HI-CELL

3.1 HUMAN-HI-C-CORPUS: A TISSUE-DIVERSE PRETRAINING RESOURCE

We collect and curate publicly available human single-cell and bulk Hi-C datasets from diverse studies and experimental protocols to construct Human-Hi-C-Corpus (Figure 1A). The single-cell component spans 17 tissues, including brain, heart, blood, breast, lung, digestive tissues, and ocular tissues, with additional diversity across brain regions, developmental stages, and cardiomyocyte differentiation. Bulk data extend this coverage to stem-cell differentiation, cell-cycle progression, and targeted perturbations. We generate pseudo-single cells by downsampling bulk Hi-C data to empirically calibrated single-cell contact depths. Data sources and sampling details are provided in Appendices E and B.

3.2 HIERARCHICAL CELL–CHROMOSOME–PATCH REPRESENTATION

HI-CELL represents each cell through three complementary views of its chromatin contacts (Figure 1B). For cell i, let $\boldsymbol{x}_i$ denote the concatenated upper triangles of its intra-chromosomal contact

maps. After chromosome-wise depth normalization, multiscale diagonal blocks are projected into *patch tokens* that encode local interaction patterns. A *chromosome token* encodes the complete contact profile of each chromosome, including interactions outside the local blocks. A *cell token* is initialized by encoding the concatenated contact profile across chromosomes. Thus, the cell representation receives genome-wide information directly, rather than relying solely on pooling local features.

Chromosome, position, and token-type embeddings retain the genomic identity and scale of these inputs. A shared Transformer jointly contextualizes all tokens through global self-attention (Figure 1C), allowing local evidence to be interpreted in chromosome and whole-cell context. Writing this encoder as $H_i = f_\theta(T(\boldsymbol{x}_i))$, where T denotes normalization and token construction, the contextualized cell token $\boldsymbol{h}_i = H_i[\mathrm{cell}]$ serves as the cellular representation. Separate decoders reconstruct local patches, chromosome maps, and the whole-cell contact profile from their corresponding tokens. This couples representation learning to contact reconstruction at each level. Architectural specifications are provided in Appendix A.

3.3 Multilevel contact reconstruction

The reconstruction objective learns the distribution of contacts within each target region. Given decoder logits z and a nonempty target count vector y, we define the count-normalized multinomial loss

$$\ell(z, y) = -\sum_j \frac{y_j}{\sum_k y_k} \log \mathrm{softmax}(z)_j. \tag{1}$$

Patch decoders predict distributions within individual blocks; chromosome and cell decoders use a separate distribution for each chromosome. This formulation models relative contact structure rather than requiring the network to reproduce sequencing depth. Let $\mathcal{L}_r(u, y)$ aggregate the reconstruction terms at level r for input u and target y. For a set of supervised levels $\mathcal{A}$, the multilevel objective is

$$\mathcal{R}_{\mathcal{A}}(u, y) = \sum_{r \in \mathcal{A}} \mathcal{L}_r(u, y), \qquad \mathcal{A} \subseteq \{\mathrm{patch}, \mathrm{chr}, \mathrm{cell}\}. \tag{2}$$

The supervised levels and input–target pairs vary across pretraining stages. Mixed structural pretraining and cross-depth refinement average the normalized losses over nonempty targets, while single-cell pretraining uses contact-count-weighted aggregation; exact reductions are given in Appendix B.2.

3.4 Three-stage progressive pretraining

The stages progressively establish a cellular representation, incorporate complementary structural supervision, and train reconstruction across contact depths (Figure 1D). The backbone and reconstruction heads are carried forward between stages.

Stage I: single-cell pretraining. The model first learns from experimentally measured single cells, reconstructing each observed profile from its encoded representation at patch, chromosome, and cell levels. This establishes a common representation of cellular contact patterns before introducing population-level structural targets.

Stage II: mixed structural pretraining. Real-cell reconstruction is combined with pseudo-single-cell-to-bulk reconstruction. For a source bulk map $\boldsymbol{b}_j$, sampling contacts without replacement at depth d produces a sparse input $S(\boldsymbol{b}_j; d)$, while $\boldsymbol{b}_j$ remains its target. Both sources supervise all three reconstruction levels with equal source weights. Real cells continue to supply observed cellular variation, while pseudo-single-cell pairs introduce higher-coverage contact structure from additional biological contexts.

Stage III: cross-depth refinement. Each real profile $\boldsymbol{x}_i$ is thinned to form $\widetilde{\boldsymbol{x}}_i$, and the original profile remains its target. Real-cell reconstruction receives greater weight, while pseudo-single-cell pairs retain auxiliary structural supervision. With $\mathcal{A}_{\mathrm{all}} = \{\mathrm{patch}, \mathrm{chr}, \mathrm{cell}\}$ and $\mathcal{A}_{\mathrm{struct}} = \{\mathrm{patch}, \mathrm{chr}\}$, the objective is

$$\begin{aligned} \mathcal{L}_{\mathrm{final}} = \lambda_{\mathrm{real}}\, \mathbb{E}_{i,\mathrm{thin}}[\mathcal{R}_{\mathcal{A}_{\mathrm{all}}}(\widetilde{\boldsymbol{x}}_i, \boldsymbol{x}_i)] \\ + \lambda_{\mathrm{pseudo}}\, \mathbb{E}_{j,d,S}[\mathcal{R}_{\mathcal{A}_{\mathrm{struct}}}(S(\boldsymbol{b}_j; d), \boldsymbol{b}_j)]\,, \end{aligned} \tag{3}$$

where $\lambda_{\text{real}} > \lambda_{\text{pseudo}} > 0$. Pseudo-single cells update the shared backbone through patch and chromosome reconstruction, while direct supervision of the cell decoder comes from real cells. This stage trains recovery from reduced contact coverage while retaining complementary population-level structural information. The final checkpoint is selected by real-cell denoising validation; loss weights and training schedules are specified in Appendix B.

3.5 Downstream applications

Cell representation. The encoder produces a cell embedding $\boldsymbol{h}_i$ for clustering and cell-type annotation, either directly or after label-free fine-tuning on the target dataset. Annotation uses a classifier trained on labeled cells with the encoder frozen.

Contact imputation. The pretrained encoder and cell decoder reconstruct a contact map from a sparse cell profile. Imputation is zero-shot: the model is applied directly to the target dataset without fine-tuning.

4 Experiments

4.1 Experimental Setup

Datasets. We evaluate transfer on six external scHi-C datasets excluded from single-cell pretraining: Ramani2017 (Ramani et al., 2017), Lee2019 (Lee et al., 2019), Kim2020 (Kim et al., 2020), Chang2024 (Chang et al., 2025), Wu2024 (Wu et al., 2024), and Zhou2024 (Zhou et al., 2024). They span cultured cell lines, human prefrontal cortex, peripheral blood mononuclear cells (PBMCs), and $CD34^+$ hematopoietic stem and progenitor cells (HSPCs), covering different experimental protocols, contact depths, and cell types. Dataset details are provided in Appendix Table 7.

Baselines. Clustering baselines are PCA, LDA, HiCRep/MDS, scHiCluster, Va3DE, and Higashi. Annotation baselines are logistic regression, random forest, and scHiClassifier, using the feature construction and implementations of scHiClassifier (Zhou & Wu, 2025). Imputation is compared with observed input, scHiCluster, and Higashi-0 and Higashi-neighbor with zero and five neighbors, respectively. Hi-Cell (sc-only) is the checkpoint after Stage I single-cell pretraining; the final model provides zero-shot or label-free fine-tuned embeddings. The same pretrained checkpoint supplies zero-shot embeddings and contact imputation. Baseline details are given in Appendix D.5.

Evaluation protocols. Leiden clustering is evaluated by normalized mutual information (NMI) and adjusted Rand index (ARI), with resolution selected to match each dataset's annotated class count. Annotation uses shared stratified five-fold cross-validation: a classifier is trained on frozen embeddings, early stopping uses training-fold validation, and scores are computed from pooled out-of-fold predictions. Following the use of downsampling to assess coverage effects in scHiCluster (Zhou et al., 2019), we evaluate imputation at fixed budgets of 10k, 25k, and 50k intra-chromosomal contacts per cell, using matched cells, inputs, references, and masks. We report held-out contact retrieval (AUPRC), held-out stratum-adjusted correlation (SCC), and leave-one-out pseudobulk SCC, averaging equally over depths and datasets. Complete protocols and additional metrics are provided in Appendix D.

4.2 Cell Clustering

Figure 2 visualizes the cell-type structure captured by six baseline representations and by zero-shot and fine-tuned Hi-Cell embeddings on the external Kim2020 dataset; corresponding comparisons for the other datasets are provided in Appendix D.1. The quantitative results in Table 1 show that the transferable structure is already present before fine-tuning: relative to sc-only pretraining, the final zero-shot model raises mean NMI from 0.5904 to 0.6423 and mean ARI from 0.5806 to 0.5925, exceeding five of six external methods in mean NMI and all six in mean ARI. On Kim2020, its zero-shot NMI/ARI of 0.9060/0.9322 closely approaches the strongest dataset-specific method without target labels or fine-tuning. Label-free fine-tuning further produces the highest six-dataset averages, reaching 0.7278 NMI and 0.7374 ARI—gains of 0.055 and 0.149 over Higashi, the strongest

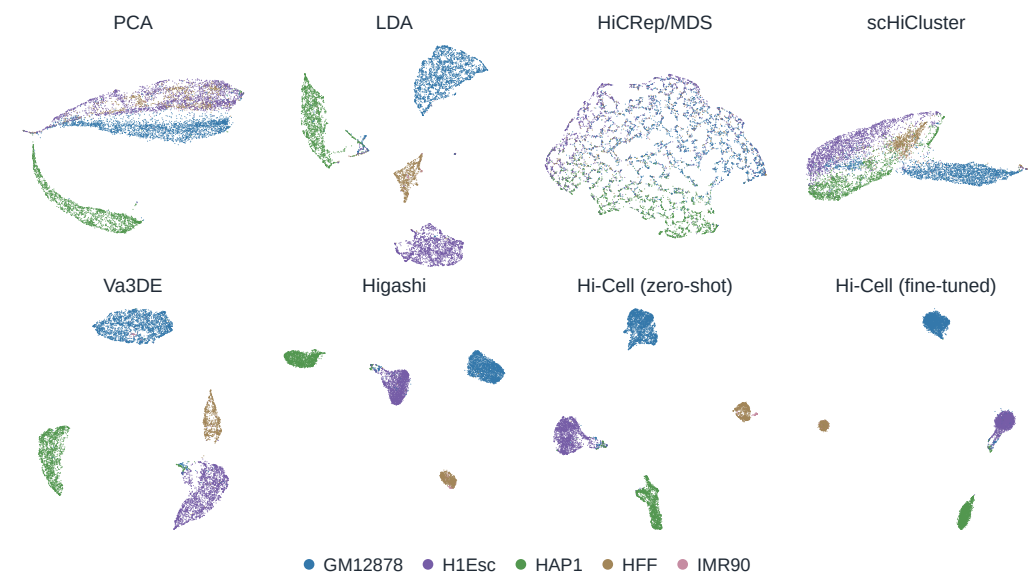


Figure 2: **Embedding comparison on the Kim2020 dataset.** Panels show UMAP projections of cell embeddings from the indicated methods. Points represent cells; colors indicate cell-line identity.

Table 1: **Leiden clustering across six external datasets.** NMI: normalized mutual information; ARI: adjusted Rand index. Mean denotes the six-dataset average. Bold and underlining mark the highest and second-highest scores per metric and column, including ties.

Method	Ramani 2017	Lee 2019	Kim 2020	Chang 2024	Wu 2024	Zhou 2024	Mean
A. NMI ↑							
PCA	0.2601	0.4411	0.5840	0.3979	0.4129	0.3035	0.3999
LDA	0.0740	0.4945	0.8780	0.2424	0.3647	0.3380	0.3986
HiCRep/MDS	0.6135	0.3846	0.2105	0.2403	0.4169	0.3811	0.3745
scHiCluster	0.2783	0.6077	0.5029	0.1324	0.8231	0.3230	0.4446
Va3DE	0.8432	0.7096	0.8417	0.4684	0.5904	0.3709	0.6374
Higashi	0.8097	0.7788	**0.9183**	0.4380	0.6751	0.4166	0.6727
HI-CELL (sc-only)	0.6530	0.7432	0.7498	0.4088	0.6054	0.3824	0.5904
HI-CELL (zero-shot)	0.7848	0.7163	0.9060	0.2554	0.7659	**0.4252**	0.6423
HI-CELL (fine-tuned)	**0.9365**	**0.7921**	0.9001	**0.4816**	**0.8599**	0.3965	**0.7278**
B. ARI ↑							
PCA	0.1733	0.2676	0.4925	0.3458	0.2693	0.1951	0.2906
LDA	0.0860	0.4270	0.9234	0.1561	0.3388	0.2300	0.3602
HiCRep/MDS	0.5180	0.1769	0.1861	0.1686	0.2269	0.2725	0.2582
scHiCluster	0.1921	0.4388	0.5205	0.1260	0.8266	0.2202	0.3874
Va3DE	0.8229	0.6732	0.8273	0.3842	0.4860	0.2833	0.5795
Higashi	0.7434	0.5948	**0.9478**	0.4241	0.4325	0.3855	0.5880
HI-CELL (sc-only)	0.6871	0.7075	0.8099	0.3623	0.6264	0.2904	0.5806
HI-CELL (zero-shot)	0.6831	0.5772	0.9322	0.2018	0.8098	0.3510	0.5925
HI-CELL (fine-tuned)	**0.9648**	**0.7793**	0.9277	**0.4574**	**0.8981**	**0.3972**	**0.7374**

external method by these averages—and leads all external baselines in nine of the twelve dataset–metric comparisons. The largest ARI margins over the best external method occur on Ramani2017 (0.142) and the human prefrontal-cortex Lee2019 dataset (0.106), while leading NMI and ARI on the Chang2024 PBMC dataset further supports transfer across protocols and biological contexts.

Table 2: **Cell-type annotation across six external datasets.** Panels report accuracy and macro-F1. Mean denotes the six-dataset average. Bold and underlining mark the highest and second-highest scores per metric and column, including ties.

Method	Ramani 2017	Lee 2019	Kim 2020	Chang 2024	Wu 2024	Zhou 2024	Mean
A. Accuracy ↑							
Logistic regression	0.9659	0.8679	0.9813	**0.8370**	0.9837	0.6714	0.8845
Random forest	0.9448	0.7029	0.8579	0.7699	0.9604	0.6057	0.8069
scHiClassifier	0.9708	0.9257	0.7724	0.8203	0.8205	0.5102	0.8033
HI-CELL (sc-only)	0.9838	**0.9424**	0.9819	0.7654	0.9883	0.6714	0.8889
HI-CELL (zero-shot)	0.9935	0.9299	0.9935	0.7179	**0.9953**	**0.7049**	0.8892
HI-CELL (fine-tuned)	**0.9951**	0.9297	**0.9954**	0.7936	**0.9953**	0.6930	**0.9004**
B. Macro-F1 ↑							
Logistic regression	0.8725	0.7642	0.8692	**0.7729**	0.9803	0.5688	0.8047
Random forest	0.7320	0.3975	0.5682	0.6019	0.9553	0.4122	0.6112
scHiClassifier	0.8929	0.8700	0.7102	0.7506	0.8162	0.4537	0.7489
HI-CELL (sc-only)	0.9765	**0.8921**	0.9518	0.6964	0.9877	0.6126	0.8529
HI-CELL (zero-shot)	0.9859	0.8782	0.9700	0.6551	**0.9954**	**0.6568**	0.8569
HI-CELL (fine-tuned)	**0.9916**	0.8740	**0.9863**	0.7297	**0.9954**	0.6340	**0.8685**

4.3 CELL-TYPE ANNOTATION

Under shared stratified five-fold cross-validation, an MLP trained on frozen zero-shot HI-CELL embeddings achieves mean macro-F1 0.8569, exceeding logistic regression, the strongest external baseline by this metric, by 5.22 percentage points (Table 2). This performance before target-dataset fine-tuning shows that contact reconstruction pretraining captures cell-type-associated structure that can be read out for annotation. Label-free fine-tuning further raises mean macro-F1 to 0.8685 and mean accuracy to 0.9004, the highest averages among the compared methods. Both zero-shot and fine-tuned embeddings outperform all three external baselines in macro-F1 on five of the six datasets. These findings demonstrate that cell-type information is accessible through a lightweight supervised readout, without cell-type labels being used to train the encoder. Together with the clustering results, they establish the value of pretraining as a shared foundation for cell grouping and cell-type annotation, connecting label-free learning of chromatin structure with cellular identity.

4.4 ZERO-SHOT CONTACT IMPUTATION

The reconstruction interface tests whether a prior learned before target analysis can recover missing contacts in new cells. Table 3 compares observed input, scHiCluster, the two native Higashi outputs, the sc-only pretraining ablation, and the final HI-CELL checkpoint under matched inputs. AUPRC measures retrieval of unobserved contacts, held-out SCC measures distance-stratified recovery of withheld counts, and leave-one-out (LOO) pseudobulk SCC measures agreement with the cell-type structural reference.

HI-CELL achieves the highest held-out SCC and LOO pseudobulk SCC on every dataset after equal averaging across depth. Across the six datasets, mean held-out SCC is 0.3478, compared with 0.1874 for Higashi-0; mean LOO pseudobulk SCC is 0.7460, compared with 0.4155 for Higashi-neighbor. In the Zhou2024 dataset example, HI-CELL recovers the broad regional contact patterns visible in the full-depth cell from a 10k-contact input (Figure 3A). These complementary comparisons identify the central reconstruction capability: recovering structural contact patterns from sparse observations without fine-tuning.

The same model achieves the highest depth-averaged AUPRC on five of six datasets. Its mean AUPRC is 0.2973, compared with 0.2633 for Higashi-neighbor. On the primary-cell datasets, held-out SCC reaches 0.1116 on the Chang2024 PBMC dataset and 0.3757 on the Zhou2024 HSPC dataset, supporting transfer beyond cultured cell lines. Figure 3B shows depth-resolved behavior, while Appendix Table 9 retains precision-at-K, rank correlations, and distance-decay error.

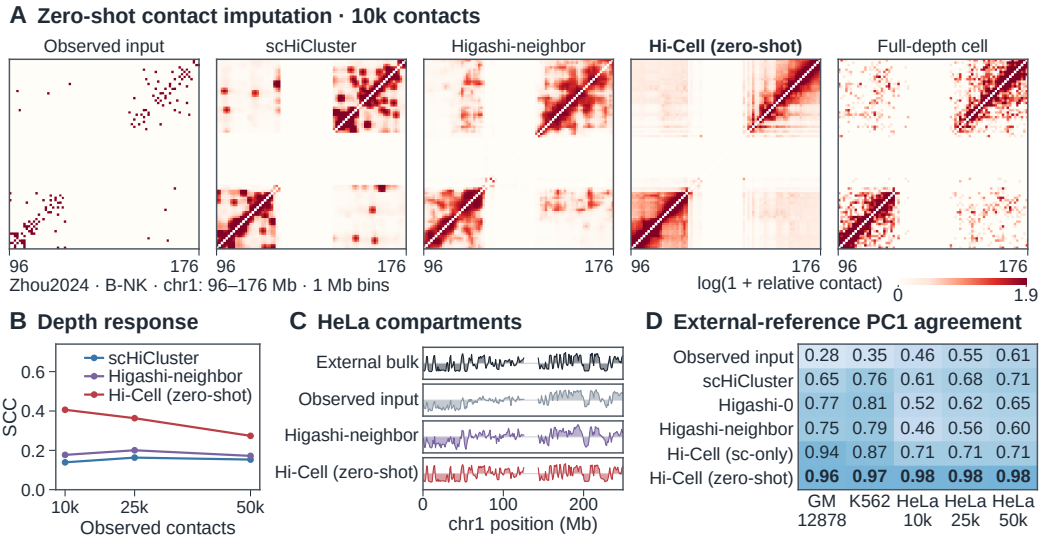


Figure 3: **Contact imputation and compartment profiles at 1 Mb resolution.** **A**, input, reconstructed, and reference contact maps for a B/NK cell from the Zhou2024 dataset at chr1:96–176 Mb. Color indicates log-scaled relative contact intensity. **B**, mean held-out SCC versus input contact count. **C**, standardized chr1 compartment profiles for HeLa pseudobulks; gaps mark excluded bins. **D**, genome-wide PC1 Pearson correlations with external bulk, shown by color and numeric labels.

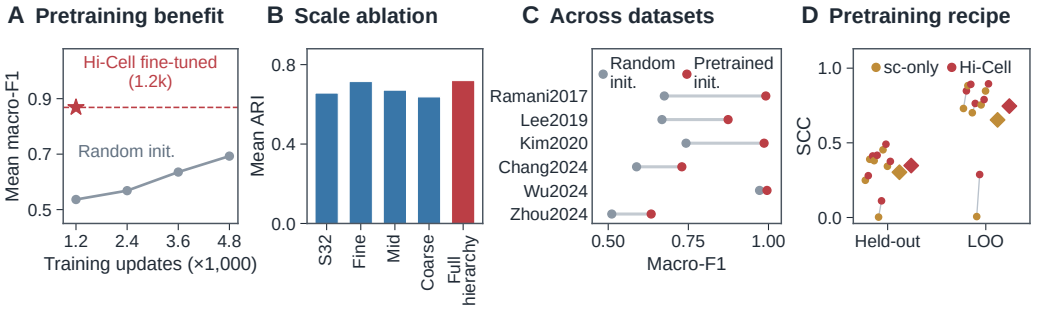


Figure 4: **Ablation analyses of pretraining and contact scales.** **A**, mean annotation macro-F1 versus target-training updates. Gray circles show random initialization; the red star and dashed line mark fine-tuned Hi-Cell. **B**, mean ARI for patch-scale variants. **C**, dataset-level macro-F1 for random (gray) and pretrained (red) initialization; lines connect the same dataset. **D**, held-out and LOO SCC for sc-only (gold) and Hi-Cell (red). Circles denote dataset means, lines pair the same dataset, and diamonds denote six-dataset means.

A/B compartment analysis. We assess chromosome-scale structure by aggregating reconstructed maps by cell type and comparing their compartment profiles with independently measured bulk Hi-C (Figure 3C–D). Distance-decay fidelity is reported in Appendix Table 9.

For GM12878 and K562, Hi-Cell reaches PC1 Pearson correlations of 0.965 and 0.974, with sign agreement of 0.939 and 0.951. For HeLa, correlation remains between 0.980 and 0.981 from 10k to 50k observed contacts, with compartment-sign F1 between 0.946 and 0.948. These results show that zero-shot imputation recovers cell-type-specific A/B compartment patterns across contact depths.

4.5 Ablation Studies

We isolate the contribution of pretraining by matching architecture and label-free target training; settings are detailed in Appendix D.4. At approximately 1,200 fine-tuning updates, the pretrained

Table 3: **Contact imputation at 1 Mb across six external datasets.** AUPRC: area under the precision–recall curve; SCC: stratum-adjusted correlation coefficient; LOO: leave-one-out. Mean denotes the six-dataset average; – marks undefined correlations. Bold and underlining mark the highest and second-highest scores per metric and column, including ties.

Method	Ramani 2017	Lee 2019	Kim 2020	Chang 2024	Wu 2024	Zhou 2024	Mean
A. AUPRC ↑							
Observed input	0.0164	0.0912	0.0644	0.0041	0.1634	0.0700	0.0683
scHiCluster	0.1232	<u>0.3452</u>	0.2295	0.0099	0.4551	0.2988	0.2436
Higashi-0	0.1400	**0.3486**	0.2398	0.0073	0.4802	0.3289	0.2575
Higashi-neighbor	0.1601	0.3221	0.2840	0.0087	0.4803	0.3244	0.2633
HI-CELL (sc-only)	<u>0.2255</u>	0.3397	<u>0.3245</u>	<u>0.0164</u>	<u>0.4881</u>	<u>0.3386</u>	<u>0.2888</u>
HI-CELL (zero-shot)	**0.2370**	0.3347	**0.3385**	**0.0325**	**0.4996**	**0.3417**	**0.2973**
B. Held-out SCC ↑							
Observed input	–	–	–	–	–	–	–
scHiCluster	0.0887	0.2636	0.1248	<u>0.0262</u>	0.2247	0.1827	0.1518
Higashi-0	0.1149	0.2864	0.1666	0.0075	0.2936	0.2553	0.1874
Higashi-neighbor	0.1192	0.2527	0.2058	0.0111	0.2739	0.2358	0.1831
HI-CELL (sc-only)	<u>0.2493</u>	<u>0.3899</u>	<u>0.3805</u>	0.0027	<u>0.4533</u>	<u>0.3429</u>	<u>0.3031</u>
HI-CELL (zero-shot)	**0.2800**	**0.4120**	**0.4166**	**0.1116**	**0.4908**	**0.3757**	**0.3478**
C. LOO pseudobulk SCC ↑							
Observed input	0.3145	0.3324	0.3627	0.0198	0.3289	0.3316	0.2817
scHiCluster	0.2398	0.3297	0.3356	0.0394	0.2361	0.2056	0.2310
Higashi-0	0.4058	0.3900	0.4523	0.0315	0.4298	0.4447	0.3590
Higashi-neighbor	0.4597	0.4553	0.5226	<u>0.0644</u>	0.4807	0.5101	0.4155
HI-CELL (sc-only)	<u>0.7300</u>	<u>0.8832</u>	<u>0.7014</u>	0.0062	<u>0.7539</u>	<u>0.8471</u>	<u>0.6536</u>
HI-CELL (zero-shot)	**0.8480**	**0.8915**	**0.7637**	**0.2877**	**0.7897**	**0.8951**	**0.7460**

model obtains macro-average NMI/ARI of 0.723/0.720 (Table 10). Figure 4A shows that its mean annotation macro-F1 of 0.8685 exceeds the random-initialization result of 0.6929 even when the latter receives four times the target-training updates, with the same ordering on all six datasets (Figure 4C). Pretraining therefore provides a more effective initialization for learning from sparse target data.

The scale ablation clarifies the representation's sensitivity to local context (Figure 4B). Fine-scale windows $\{8, 16\}$ retain a mean ARI of 0.7146, close to 0.7195 for the full hierarchy, compared with 0.6567 for a single scale and 0.6372 for coarse windows. Thus fine-grained contact patterns provide an effective basis for cellular identity. The full hierarchy retains these local features while also providing chromosome- and cell-level context for reconstruction. The final pretraining recipe additionally improves mean held-out and LOO pseudobulk SCC over the sc-only variant (Table 3; Figure 4D), connecting the representation analysis to the reconstruction interface.

5 CONCLUSION

We introduced HI-CELL, a foundation model for single-cell 3D genome analysis that learns a transferable structural prior from real and pseudo-single cells. A shared pretrained backbone supports cell representation and contact reconstruction. Evaluations across six external datasets demonstrate effective cell clustering and annotation, together with zero-shot contact imputation. Controlled comparisons with random initialization establish the contribution of pretraining to downstream representation learning. These results show how accumulated Hi-C data can support reusable models for studying cellular identity and chromatin organization from sparse single-cell measurements.

Reproducibility Statement

An anonymized code archive containing the model implementation, training configurations, embedding and imputation scripts, and usage instructions is provided with this submission as supplementary material. The pretrained Hi-Cell weights will be publicly released. Model and training details are documented in Appendices A–C, evaluation protocols in Appendix D, and data sources in Appendix E.

References

Lei Chang, Yang Xie, Brett Taylor, Zhaoning Wang, Jiachen Sun, Ethan J. Armand, Shreya Mishra, Jie Xu, Melodi Tastemel, Audrey Lie, Zane A. Gibbs, Hannah S. Indralingam, Tuyet M. Tan, Rafael Bejar, Clark C. Chen, Frank B. Furnari, Ming Hu, and Bing Ren. Droplet Hi-C enables scalable, single-cell profiling of chromatin architecture in heterogeneous tissues. *Nature Biotechnology*, 43(10):1694–1707, 2025. ISSN 1546-1696. doi: 10.1038/s41587-024-02447-1. URL https://doi.org/10.1038/s41587-024-02447-1.

Xiaoyang Chen, Keyi Li, Xuejian Cui, Zian Wang, Qun Jiang, Jiacheng Lin, Zhen Li, Zijing Gao, Hairong Lv, and Rui Jiang. EpiAgent: foundation model for single-cell epigenomics. *Nature Methods*, 22(11):2316–2327, 2025a. ISSN 1548-7105. doi: 10.1038/s41592-025-02822-z. URL https://doi.org/10.1038/s41592-025-02822-z.

Zeyuan Johnson Chen, Sankha Subhra Das, Asha Kar, Seung Hyuk T. Lee, Kevin D. Abuhanna, Marcus Alvarez, Mihir G. Sukhatme, Zitian Wang, Kyla Z. Gelev, Matthew G. Heffel, Yi Zhang, Oren Avram, Elior Rahmani, Sriram Sankararaman, Markku Laakso, Sini Heinonen, Hilkka Peltoniemi, Eran Halperin, Kirsi H. Pietiläinen, Chongyuan Luo, and Päivi Pajukanta. Single-cell DNA methylome and 3D genome atlas of human subcutaneous adipose tissue. *Nature Genetics*, 57(9):2238–2249, 2025b. ISSN 1546-1718. doi: 10.1038/s41588-025-02300-4. URL https://doi.org/10.1038/s41588-025-02300-4.

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. ISSN 1548-7105. doi: 10.1038/s41592-024-02201-0. URL https://doi.org/10.1038/s41592-024-02201-0.

Zoe L. Grant, Shuzhen Kuang, Shu Zhang, Abraham J. Horrillo, Zhe Chen, Kavitha S. Rao, Cemre Celen, Vasumathi Kameswaran, Carine Joubran, Pik Ki Lau, Keyi Dong, Bing Yang, Weronika M. Bartosik, Nathan R. Zemke, Bing Ren, Deepak Srivastava, Irfan S. Kathiriya, Katherine S. Pollard, and Benoit G. Bruneau. Dose-dependent sensitivity of human three-dimensional chromatin to a heart disease–linked transcription factor. *Science*, 393(6809):eadv5434, 2026. doi: 10.1126/science.adv5434. URL https://www.science.org/doi/abs/10.1126/science.adv5434.

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. ISSN 1548-7105. doi: 10.1038/s41592-024-02305-7. URL https://doi.org/10.1038/s41592-024-02305-7.

Matthew G. Heffel, Jingtian Zhou, Yi Zhang, Dong-Sung Lee, Kangcheng Hou, Oier Pastor-Alonso, Kevin D. Abuhanna, Joseph Galasso, Colin Kern, Chu-Yi Tai, Carlos Garcia-Padilla, Mahsa Nafisi, Yi Zhou, Anthony D. Schmitt, Terence Li, Maximilian Haeussler, Brittney Wick, Martin Jinye Zhang, Fangming Xie, Ryan S. Ziffra, Eran A. Mukamel, Eleazar Eskin, Tomasz J. Nowakowski, Jesse R. Dixon, Bogdan Pasaniuc, Joseph R. Ecker, Quan Zhu, Bogdan Bintu, Mercedes F. Paredes, and Chongyuan Luo. Temporally distinct 3D multi-omic dynamics in the developing human brain. *Nature*, 635(8038):481–489, 2024. ISSN 1476-4687. doi: 10.1038/s41586-024-08030-7. URL https://doi.org/10.1038/s41586-024-08030-7.

Hyeon-Jin Kim, Galip Gürkan Yardımcı, Giancarlo Bonora, Vijay Ramani, Jie Liu, Ruolan Qiu, Choli Lee, Jennifer Hesson, Carol B Ware, Jay Shendure, et al. Capturing cell type-specific chromatin compartment patterns by applying topic modeling to single-cell Hi-C data. *PLoS computational biology*, 16(9):e1008173, 2020.

Dong-Sung Lee, Chongyuan Luo, Jingtian Zhou, Sahaana Chandran, Angeline Rivkin, Anna Bartlett, Joseph R. Nery, Conor Fitzpatrick, Carolyn O'Connor, Jesse R. Dixon, and Joseph R. Ecker. Simultaneous profiling of 3D genome structure and DNA methylation in single human cells. *Nature Methods*, 16(10):999–1006, 2019. ISSN 1548-7105. doi: 10.1038/s41592-019-0547-z. URL https://doi.org/10.1038/s41592-019-0547-z.

Dylan Plummer, Xiuyuan Lang, Shanshan Zhang, Yan Li, Jing Li, and Fulai Jin. A comprehensive benchmark of single-cell Hi-C embedding tools. *Nature Communications*, 16(1):9119, 2025. ISSN 2041-1723. doi: 10.1038/s41467-025-64186-4. URL https://doi.org/10.1038/s41467-025-64186-4.

Vijay Ramani, Xinxian Deng, Ruolan Qiu, Kevin L Gunderson, Frank J Steemers, Christine M Disteche, William S Noble, Zhijun Duan, and Jay Shendure. Massively multiplex single-cell Hi-C. *Nature Methods*, 14(3):263–266, 2017. ISSN 1548-7105. doi: 10.1038/nmeth.4155. URL https://doi.org/10.1038/nmeth.4155.

Sarah B. Reiff, Andrew J. Schroeder, Koray Kırlı, Andrea Cosolo, Clara Bakker, Luisa Mercado, Soohyun Lee, Alexander D. Veit, Alexander K. Balashov, Carl Vitzthum, William Ronchetti, Kent M. Pitman, Jeremy Johnson, Shannon R. Ehmsen, Peter Kerpedjiev, Nezar Abdennur, Maxim Imakaev, Serkan Utku Öztürk, Uğur Çamoğlu, Leonid A. Mirny, Nils Gehlenborg, Burak H. Alver, and Peter J. Park. The 4D Nucleome Data Portal as a resource for searching and visualizing curated nucleomics data. *Nature Communications*, 13(1):2365, 2022. ISSN 2041-1723. doi: 10.1038/s41467-022-29697-4. URL https://doi.org/10.1038/s41467-022-29697-4.

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(7965):616–624, 2023. ISSN 1476-4687. doi: 10.1038/s41586-023-06139-9. URL https://doi.org/10.1038/s41586-023-06139-9.

Wei Tian, Jingtian Zhou, Anna Bartlett, Qiurui Zeng, Hanqing Liu, Rosa G. Castanon, Mia Kenworthy, Jordan Altshul, Cynthia Valadon, Andrew Aldridge, Joseph R. Nery, Huaming Chen, Jiaying Xu, Nicholas D. Johnson, Jacinta Lucero, Julia K. Osteen, Nora Emerson, Jon Rink, Jasper Lee, Yang E. Li, Kimberly Siletti, Michelle Liem, Naomi Claffey, Carolyn O'Connor, Anna Marie Yanny, Julie Nyhus, Nick Dee, Tamara Casper, Nadiya Shapovalova, Daniel Hirschstein, Song-Lin Ding, Rebecca Hodge, Boaz P. Levi, C. Dirk Keene, Sten Linnarsson, Ed Lein, Bing Ren, M. Margarita Behrens, and Joseph R. Ecker. Single-cell DNA methylation and 3D genome architecture in the human brain. *Science*, 382(6667):eadf5357, 2023. doi: 10.1126/science.adf5357. URL https://www.science.org/doi/abs/10.1126/science.adf5357.

Xiao Wang, Yuanyuan Zhang, Suhita Ray, Anupama Jha, Tangqi Fang, Shengqi Hang, Sergei Doulatov, William S. Noble, and Sheng Wang. A generalizable Hi-C foundation model for chromatin architecture, single-cell and multiomics analysis across species. *Nature Methods*, 23(7):1334–1348, 2026. ISSN 1548-7105. doi: 10.1038/s41592-026-03097-8. URL https://doi.org/10.1038/s41592-026-03097-8.

Honggui Wu, Jiankun Zhang, Fanchong Jian, Jinxin Phaedo Chen, Yinghui Zheng, Longzhi Tan, and X. Sunney Xie. Simultaneous single-cell three-dimensional genome and gene expression profiling uncovers dynamic enhancer connectivity underlying olfactory receptor choice. *Nature Methods*, 21(6):974–982, 2024. ISSN 1548-7105. doi: 10.1038/s41592-024-02239-0. URL https://doi.org/10.1038/s41592-024-02239-0.

Xiaoqing Wu, Zian Wang, Rui Jiang, and Xiaoyang Chen. Hi-Cformer enables multiscale chromatin contact map modeling for single-cell Hi-C data analysis. *Science Advances*, 12(35):eaeg0134, 2026. doi: 10.1126/sciadv.aeg0134. URL https://www.science.org/doi/10.1126/sciadv.aeg0134.

Tao Yang, Feipeng Zhang, Galip Gürkan Yardımcı, Fan Song, Ross C. Hardison, William Stafford Noble, Feng Yue, and Qunhua Li. HiCRep: assessing the reproducibility of Hi-C data using a stratum-adjusted correlation coefficient. *Genome Research*, 27(11):1939–1949, 2017.

doi: 10.1101/gr.220640.117. URL http://genome.cshlp.org/content/genome/27/11/1939.

Ying Yuan, Pooja Biswas, Nathan R. Zemke, Kelsey Dang, Yue Wu, Matteo D'Antonio, Yang Xie, Qian Yang, Keyi Dong, Pik Ki Lau, Daofeng Li, Chanrung Seng, Weronika Bartosik, Justin Buchanan, Lin Lin, Ryan Lancione, Kangli Wang, Seoyeon Lee, Zane Gibbs, Bing Yang, Joseph R. Ecker, Kelly Frazer, Ting Wang, Sebastian Preissl, Allen Wang, Radha Ayyagari, and Bing Ren. Single-cell analysis of the epigenome and 3D chromatin architecture in the human retina. *Science Advances*, 12(30):eadv9162, 2026. doi: 10.1126/sciadv.adv9162. URL https://www.science.org/doi/abs/10.1126/sciadv.adv9162.

Ruochi Zhang, Tianming Zhou, and Jian Ma. Multiscale and integrative single-cell Hi-C analysis with Higashi. *Nature Biotechnology*, 40(2):254–261, 2022. ISSN 1546-1696. doi: 10.1038/s41587-021-01034-y. URL https://doi.org/10.1038/s41587-021-01034-y.

Ye Zheng, Siqi Shen, and Sündüz Keleş. Normalization and de-noising of single-cell Hi-C data with BandNorm and scVI-3D. *Genome Biology*, 23(1):222, 2022. ISSN 1474-760X. doi: 10.1186/s13059-022-02774-z. URL https://doi.org/10.1186/s13059-022-02774-z.

Jingtian Zhou, Jianzhu Ma, Yusi Chen, Chuankai Cheng, Bokan Bao, Jian Peng, Terrence J. Sejnowski, Jesse R. Dixon, and Joseph R. Ecker. Robust single-cell Hi-C clustering by convolution- and random-walk–based imputation. *Proceedings of the National Academy of Sciences*, 116(28):14011–14018, 2019. doi: 10.1073/pnas.1901423116. URL https://www.pnas.org/doi/abs/10.1073/pnas.1901423116.

Jingtian Zhou, Yue Wu, Hanqing Liu, Wei Tian, Rosa G. Castanon, Anna Bartlett, Zuolong Zhang, Guocong Yao, Dengxiaoyu Shi, Ben Clock, Samantha Marcotte, Joseph R. Nery, Michelle Liem, Naomi Claffey, Lara Boggeman, Cesar Barragan, Rafael Arrojo e Drigo, Annika K. Weimer, Minyi Shi, Johnathan Cooper-Knock, Sai Zhang, Michael P. Snyder, Sebastian Preissl, Bing Ren, Carolyn O'Connor, Shengbo Chen, Chongyuan Luo, Jesse R. Dixon, and Joseph R. Ecker. Human body single-cell atlas of three-dimensional genome organization and DNA methylation. *Science*, 393(6809):eadx0673, 2026. doi: 10.1126/science.adx0673. URL https://www.science.org/doi/abs/10.1126/science.adx0673.

Tianming Zhou, Ruochi Zhang, Deyong Jia, Raymond T. Doty, Adam D. Munday, Daniel Gao, Li Xin, Janis L. Abkowitz, Zhijun Duan, and Jian Ma. GAGE-seq concurrently profiles multiscale 3D genome organization and gene expression in single cells. *Nature Genetics*, 56(8):1701–1711, 2024. ISSN 1546-1718. doi: 10.1038/s41588-024-01745-3. URL https://doi.org/10.1038/s41588-024-01745-3.

Xiangfei Zhou and Hao Wu. scHiClassifier: a deep learning framework for cell type prediction by fusing multiple feature sets from single-cell Hi-C data. *Briefings in Bioinformatics*, 26(1):bbaf009, 2025.

A CONTACT REPRESENTATION AND MODEL ARCHITECTURE

This section specifies the representation, information flow, and reconstruction interfaces of HI-CELL. Appendix B gives the training objective and optimization schedule; Appendix C specifies label-free fine-tuning. All dimensions below refer to the backbone used for the main results.

A.1 CHROMOSOME-INDEXED CONTACT REPRESENTATION

The model uses the GRCh38 chromosome grid at 1 Mb resolution, with chromosomes 1–22 and X. Only intra-chromosomal contacts are represented. Let $X_{ic} \in \mathbb{R}_{\geq 0}^{n_c \times n_c}$ be the symmetric contact matrix of cell i on chromosome c. We store its upper triangle, including the diagonal, as $\boldsymbol{x}_{ic} = \mathrm{vech}_{\mathrm{upper}}(X_{ic})$ and concatenate chromosomes to obtain

$$\boldsymbol{x}_i = [\boldsymbol{x}_{i1}; \ldots; \boldsymbol{x}_{iC}], \qquad C = 23, \qquad W = \sum_{c=1}^{C} \frac{n_c(n_c+1)}{2} = 238{,}771. \tag{4}$$

Here, genome-wide refers to the collection of cis maps, not a matrix containing inter-chromosomal contacts. Each contact is stored once; off-diagonal entries are mirrored only when a symmetric local matrix is required. The chromosome order is fixed by the released index arrays rather than reconstructed by alphabetical sorting. Each dataset is checked against those arrays before model inference.

Normalization and valid support. Encoder inputs are normalized chromosome-wise: $\bar{\boldsymbol{x}}_{ic} = \boldsymbol{x}_{ic} / \max(d_{ic}, \epsilon)$, with $\epsilon = 10^{-8}$. Native canonical inputs store symmetric chromosome depth, $d_{ic} = \sum_a X_{ic,aa} + 2\sum_{a<b} X_{ic,ab}$. For the binomially thinned real-cell training views, the saved training implementation recomputes d_{ic} as the sum of the thinned packed upper triangle. The data loader supplies this depth alongside the contact vector; the same supplied depth is used for the patch, chromosome, and cell encoders. An empty chromosome therefore has a zero-valued normalized input. Reconstruction targets remain raw counts and are normalized inside the loss, separately from encoder normalization.

The largest chromosome has 249 bins. Chromosome encoders share a 249×249 upper-triangular coordinate template, with shorter chromosomes occupying their valid coordinates. The implementation selects the corresponding weight columns, which is equivalent to zero-padding without materializing a padded matrix. Padded positions are excluded from chromosome reconstruction. A zero within the valid chromosome grid denotes an unobserved contact, not padding. Diagonal entries are retained during representation learning; the imputation evaluation excludes them.

A.2 THREE COMPLEMENTARY TOKEN FAMILIES

Multiscale patch tokens. For each scale $s \in \mathcal{S} = \{8, 16, 32, 64, 128\}$, we extract non-overlapping $s \times s$ blocks along the diagonal of each chromosome, using stride s. Each full block is flattened and linearly projected to $D = 256$ dimensions, followed by layer normalization:

$$t_{icsp}^{\mathrm{patch}} = \mathrm{LN}\big(E_s \,\mathrm{vec}(\bar{X}_{ic}[ps : (p+1)s,\, ps : (p+1)s]) + b_s\big)\,. \tag{5}$$

The projection is shared across chromosomes at the same scale. Both halves of a symmetric block are included in its s^2-dimensional input. Only complete blocks are extracted: $p = 0, \ldots, \lfloor n_c/s \rfloor - 1$. Terminal bins outside a complete block still enter the chromosome and cell encoders. Off-diagonal contacts outside these local blocks likewise remain available through the chromosome and cell pathways.

Chromosome tokens. A chromosome token encodes the entire normalized upper triangle, rather than pooling patch tokens:

$$t_{ic}^{\mathrm{chr}} = \mathrm{LN}\left(E_{\mathrm{chr}} P_c \bar{\boldsymbol{x}}_{ic} + b_{\mathrm{chr}}\right), \tag{6}$$

where P_c embeds valid coordinates into the shared upper-triangular template. This branch provides chromosome-wide context, including contacts not covered by diagonal patches.

Table 4: **Architecture specification.** Arrows indicate input and output dimensions.

Component	Specification
Patch encoder	One $s^2 \to 256$ projection per scale, followed by layer normalization
Chromosome encoder	Shared $31{,}125 \to 256$ projection on the padded upper triangle, followed by layer normalization
Cell encoder	$238{,}771 \to 2048 \to 512 \to 256$; GELU hidden activations and output layer normalization
Transformer	12 blocks; 8 heads; head width 32; feed-forward width 1024; attention/projection/MLP dropout zero
Token output head	$256 \to 256 \to 256$, GELU, layer normalization
Patch decoder	One $256 \to s^2$ linear head per scale
Chromosome decoder	Shared $256 \to 31{,}125$ linear head; select valid coordinates per chromosome
Cell decoder	$256 \to 238{,}771$ linear head; separate probability normalization within each chromosome
Embedding output	Final cell-token feature, dimension 256

Cell token. The initial cell token is data-dependent:

$$t_i^{\text{cell}} = \text{LN}\left(E_{\text{cell}}([\bar{\boldsymbol{x}}_{i1}; \ldots; \bar{\boldsymbol{x}}_{iC}])\right). \tag{7}$$

E_{cell} is an MLP with widths $W \to 2048 \to 512 \to 256$ and GELU between successive linear layers. Thus the cell token already receives the full cis-contact profile before attention; it is not a constant classification token appended to local features.

A.3 Joint contextualization and output heads

The sequence begins with the cell token, followed by chromosome segments; each segment contains its chromosome token and the patch tokens ordered by scale and genomic position. Additive learned embeddings identify chromosome, within-scale patch position, and token type. The seven token types distinguish cell, chromosome, and the five patch scales. Cell and chromosome tokens use position index zero; patch positions begin at one within each scale. The cell token has no chromosome embedding.

The complete sequence contains 707 tokens: one cell token, 23 chromosome tokens, and respectively 370, 180, 84, 36, and 13 patch tokens at the five scales. A stack of 12 pre-normalization Transformer blocks uses eight attention heads, hidden width 256, and a feed-forward expansion factor of four. For block ℓ,

$$U^\ell = H^{\ell-1} + \text{MHA}(\text{LN}(H^{\ell-1})), \tag{8}$$

$$H^\ell = U^\ell + \text{FFN}(\text{LN}(U^\ell)). \tag{9}$$

Attention is global across all tokens, with no chromosome-block attention mask. The hierarchy refers to the three representational levels, not a restriction to sequential bottom-up pooling. A shared $256 \to 256 \to 256$ token MLP with GELU, followed by layer normalization, produces the decoder-input tokens. The contextualized cell token at this output is the 256-dimensional representation used in every embedding experiment.

All projections include biases, and all three reconstruction heads operate on contextualized tokens (Table 4). The instantiated model contains 594,310,984 parameters. The dense cell-input encoder, including its output normalization, accounts for 490,185,984 parameters, and the cell decoder accounts for 61,364,147. The total therefore includes the contact input/output interfaces rather than describing the Transformer stack alone.

Reconstruction interface. Imputation uses the cell decoder, not an average of patch and chromosome predictions. Writing its output as chromosome-specific logits z_{ic}, the probability assigned to valid position j is

$$p_{icj} = \frac{\exp(z_{icj})}{\sum_{k \in \mathcal{I}_c} \exp(z_{ick})}, \qquad \widehat{x}_{icj} = m_{ic} p_{icj}, \qquad m_{ic} = \sum_{j \in \mathcal{I}_c} x_{icj}^{\text{input}}. \tag{10}$$

Here $\mathcal{I}_c$ contains the chromosome's valid upper-triangular coordinates. The output adapter can retain probabilities or input-depth-rescaled scores; benchmark manifests fix the representation supplied to the evaluator. Count-like scores use the observed packed chromosome depth, never the full-depth target. The same frozen checkpoint is used at every evaluation depth, without blending its output with a reference map.

B Complementary Real and Pseudo-Single-Cell Pretraining

B.1 Real-cell denoising and pseudo-single-cell construction

For real-cell denoising, a keep probability $q \sim \mathcal{U}(0.25, 0.75)$ is sampled for each training minibatch, and each packed count is independently thinned:

$$\widetilde{x}_{icj} \sim \mathrm{Binomial}(x_{icj}, q). \tag{11}$$

The unthinned cell remains the reconstruction target. This is contact-count thinning, not token masking: cross-depth refinement (the final phase) uses a zero token-mask ratio and preserves the token sequence.

Pseudo-single cells are generated from canonical bulk cis-contact maps by sampling individual contacts without replacement. The empirical depth pool is formed by rounding the product of a cell's QC total-contact count and cis fraction, retaining values between 5,000 and 50,000. A requested depth is drawn from that pool and used to sample a sparse contact vector from its bulk parent; repeated contacts at the same bin pair retain their integer multiplicity. The source bulk, rather than the pseudo-single cell, is the reconstruction target. The saved generator preserves the parent identifier and sampling provenance for each pseudo-single cell.

Calibration uses scalar QC summaries from the Chen2025, Grant2025, Heffel2024, Kim2020, Lee2019, Ramani2017, and Wu2024 datasets. This use of aggregate depth information is distinct from including their contact profiles in single-cell pretraining: no calibration-cell contact coordinates or cell-type labels enter the generator. The six external benchmark datasets are excluded from the single-cell pretraining pool. The cis fraction calibrates sampled depth; it is not an additional trans-contact prediction target.

Pseudo-single cells inherit their bulk parent's split. A deterministic parent-level split uses a validation fraction of 0.1; pseudo-single cells from one bulk experiment cannot be divided between training and validation. During the final phase, training pseudo minibatches are sampled hierarchically by cell-type group, then bulk parent, then its pseudo batch, so that the number of pseudo-single cells from an experiment does not alone determine its frequency. Cell-type groups are sampling metadata, not prediction labels. Real-cell train/validation assignments are inherited from the fixed canonical batch partition.

B.2 Count-normalized reconstruction objectives

For decoder logits $z \in \mathbb{R}^K$ and a nonempty target $y \in \mathbb{R}^K_{\geq 0}$, define

$$\ell(z, y) = -\sum_{j=1}^{K} \frac{y_j}{\sum_k y_k} \log \mathrm{softmax}(z)_j. \tag{12}$$

This is the multinomial negative log-likelihood normalized by target contact count, omitting parameter-independent terms. Empty targets are excluded from the reduction, rather than assigned an artificial distribution. Normalization prevents a dense bulk target from receiving more loss weight solely because it contains more contacts.

The patch objective averages Equation 12 over all nonempty patch targets in the minibatch, across chromosomes and scales. A full symmetric $s \times s$ patch is the target support for scale s. The chromosome objective averages over nonempty cell–chromosome targets on their valid upper triangles. Although the cell decoder emits a concatenated genome-wide vector, its loss applies a separate softmax and count normalization to each chromosome slice, then averages over nonempty cell–chromosome targets. Thus the cell loss does not use one global softmax across chromosomes. Each branch is normalized before the three losses are combined.

Table 5: **Reconstruction supervision during cross-depth refinement.** Rows specify each input and its reconstruction target. "Yes" indicates a decoder level receiving direct reconstruction supervision; a dash indicates no direct loss at that level.

Input	Target	Patch	Chromosome	Cell
Thinned real cell	Original cell	yes	yes	yes
Pseudo-single cells	Source bulk	yes	yes	—

Table 6: **Optimization settings for the three pretraining stages.** Completed budget gives the total training duration; selected checkpoint gives the epoch or update used for subsequent training or evaluation. Epoch counts are one-based completed epochs, and updates are optimizer steps. Real/pseudo entries denote real single cells and pseudo-single cells; chr denotes chromosome-level reconstruction. A dash indicates an unused input or loss.

Setting	Single-cell pretraining	Mixed structural pretraining	Cross-depth refinement
Learning rate	5×10^{-4}	10^{-4}	Peak 2×10^{-5}
Real / pseudo batch	256 / —	256 / 128	Global 256 / 128
Completed budget	167 epochs	125,600 updates	30,720 updates
Selected checkpoint	Epoch 157	Update 125,600	Update 28,800
Real / pseudo weights	1 / 0	1 / 1	1.8 / 0.2
Pseudo decoder losses	—	Patch + chr + cell	Patch + chr
Selection criterion	Real-cell validation reconstruction	Equal real/pseudo validation mixture	Real-cell multidepth denoising

The final real-cell and pseudo-single-cell objectives are

$$\mathcal{L}_{\text{real}} = \mathcal{L}_{\text{patch}}^{\text{real}} + \mathcal{L}_{\text{chr}}^{\text{real}} + \mathcal{L}_{\text{cell}}^{\text{real}}, \tag{13}$$

$$\mathcal{L}_{\text{pseudo}} = \mathcal{L}_{\text{patch}}^{\text{pseudo}} + \mathcal{L}_{\text{chr}}^{\text{pseudo}}. \tag{14}$$

The implemented final objective is $1.8\mathcal{L}_{\text{real}} + 0.2\mathcal{L}_{\text{pseudo}}$, corresponding to $\lambda_{\text{real}} = 1.8$ and $\lambda_{\text{pseudo}} = 0.2$ in Equation 3. These are fixed loss coefficients, not sampling probabilities or inferred uncertainty scores.

Supervision at the appropriate level. Real cells directly supervise all three reconstruction levels. Bulk-derived targets supervise patch and chromosome structure, but do not directly supervise the cell decoder in the final phase. Pseudo-single cells still update the shared encoders and Transformer through contextual attention. This design uses population contact structure as auxiliary supervision while retaining an explicit real-cell reconstruction objective for the cell-level output.

B.3 Optimization schedule and checkpoint selection

Training first establishes a single-cell representation, then introduces bulk-derived structural supervision, and finally performs cross-depth refinement with the asymmetric objective above (Figure 1D). These phases share the same contact representation and backbone. The sc-only comparison is the checkpoint after Stage I single-cell pretraining; final HI-CELL uses the checkpoint selected after cross-depth refinement.

Single-cell pretraining uses the count-weighted multinomial reduction of its original trainer; mixed structural pretraining and cross-depth refinement use the normalized nonempty-target reduction in Equation 12. Mixed structural pretraining reconstructs unthinned real inputs and source-bulk targets. The final phase introduces real-cell thinning together with the asymmetric supervision scheme, and is therefore specified as a complete training recipe.

Final-phase optimization uses AdamW with zero weight decay, default moments $(\beta_1, \beta_2) = (0.9, 0.999)$, numerical stabilizer 10^{-8}, FP32 computation, and gradient-norm clipping at 1.0. Four-device data parallelism uses local real/pseudo batch sizes 64/32 and one synchronized update per pair of domain batches. The learning rate warms up over 256 updates and then decays by a co-

sine schedule to one tenth of its peak. Training comprises 48 epochs of 640 updates. There is no automatic domain-weight calibration in this run.

Checkpoint selection uses no downstream labels. At each validation event, real-cell denoising losses are evaluated at fixed keep probabilities 0.25, 0.50, and 0.75 and averaged equally, using at most 32 batches per rank and depth. The checkpoint minimizing this criterion is retained; the selected update is 28,800. This frozen checkpoint is the common source for zero-shot embeddings, zero-shot imputation, and target-dataset fine-tuning.

Algorithm: final-phase training.

1. Initialize the backbone and reconstruction heads from mixed structural pretraining; fix the real-cell and bulk-parent train/validation partitions.
2. Draw a real-cell minibatch and a hierarchically sampled pseudo-single-cell minibatch with their source-bulk targets.
3. Draw $q \sim \mathcal{U}(0.25, 0.75)$, binomially thin the real counts, and recompute their input depths. Keep the original real counts as targets.
4. Encode both domains with shared parameters and global self-attention. Compute patch, chromosome, and cell losses for real cells, and patch and chromosome losses for pseudo-single cells.
5. Backpropagate $1.8\mathcal{L}_{\text{real}} + 0.2\mathcal{L}_{\text{pseudo}}$, clip the gradient norm, and apply one AdamW update and one scheduler step.
6. After each epoch, compute the fixed multidepth real-cell validation objective and retain its best checkpoint. Stop after the prescribed update budget.

C Label-Free Fine-Tuning and Controlled Ablations

C.1 One label-free fine-tuning rule

Each target dataset starts independently from the same final pretrained checkpoint. All backbone and decoder parameters are trainable; fine-tuning reconstructs the original target contact maps without thinning or token masking. Patch, chromosome, and cell losses have coefficients $1 : 1 : 1$ and use the normalized reduction above. No cell-type label, consistency anchor, or relation-preservation penalty enters this objective.

The optimizer is AdamW with learning rate 10^{-5}, weight decay zero, batch size 128, FP32 arithmetic, gradient accumulation one, and gradient clipping at 1.0. Canonical source batches are split deterministically into 90% training and 10% reconstruction validation. The stopping budget is fixed at approximately 1,200 optimizer updates, rounded to a complete epoch for each dataset. Validation is logged but does not choose the reported stopping point. No model is initialized from a different target dataset's adapted weights.

Fine-tuning is dataset-level and label-free: its split is independent of the supervised classifier folds. Thus cells later assigned to a classifier test fold may have participated in label-free fine-tuning, but their labels do not train or select the encoder. After fine-tuning, embeddings are frozen before the supervised readout. Zero-shot embeddings instead use the pretrained encoder without fine-tuning. Both settings use the same downstream evaluation contract.

C.2 What each ablation changes

The random-initialization comparison uses the same full architecture, target data, reconstruction objective, optimizer, and approximately 1,200-update budget as pretrained fine-tuning. It uses no pretraining data or pseudo-single-cell minibatches during target training. The longer random-initialization control increases the target-training budget to approximately 4,800 updates while keeping the other settings fixed. Update comparisons therefore measure training on the target dataset, not total pretraining-plus-fine-tuning compute.

Scale ablations retain $\{32\}$, $\{8, 16\}$, $\{16, 32, 64\}$, $\{64, 128\}$, or all five scales. The retained scales use their corresponding pretrained patch encoder, decoder, and scale-type embeddings; chromosome

Table 7: **Datasets used for external evaluation.** Repr. and Recon. give the numbers of cells used for representation and reconstruction evaluation, respectively; Classes gives the number of annotated cell types. PBMC denotes peripheral blood mononuclear cells; HSPC denotes hematopoietic stem and progenitor cells.

Dataset	Repr.	Recon.	Classes	Biological setting
Ramani2017	616	64	4	Cell lines
Lee2019	4,238	3,896	14	Human prefrontal cortex
Kim2020	7,894	97	5	Cell lines
Chang2024	4,502	84	7	Human PBMC
Wu2024	429	428	4	Cell lines
Zhou2024	837	837	6	$CD34^+$ HSPC
Total	18,516	5,406		

and cell pathways remain present. These experiments vary which local scales are available during fine-tuning and evaluation, rather than pretraining a new backbone for every scale subset. The sc-only versus final-recipe reconstruction comparison uses the same downstream inputs and evaluator and contrasts the two pretrained checkpoints specified in Table 6.

D Evaluation Protocols

D.1 Visualization protocol

UMAPs use Scanpy with 20 neighbors, minimum distance 0.1, and Euclidean distance. Saved cell identities and row-order records align all eight methods to the same cells and labels within each dataset. UMAP is applied directly to each method's native embeddings. Labels are used only for coloring, never projection; all cells are retained, and isotropic display scaling preserves within-panel geometry. Each panel is independently projected; absolute coordinates are not shared across methods. A fixed random point-drawing order is shared across panels. Figure 3A uses a 10k-contact input from one B/NK cell in the Zhou2024 dataset. Contact examples are previously locked illustrative selections, not population-level results. Each chr1 matrix is normalized by its mean over off-diagonal offsets 3–50, transformed by $\log(1 + x)$, and displayed using one pooled 98.5th-percentile color limit; the diagonal is masked identically. HeLa compartment tracks are cell-type pseudobulk profiles at 10k contacts per cell, standardized per track in a shared external-bulk PCA basis. Excluded bins remain gaps. Signs are reference-oriented, not independently assigned biological A/B labels. Figure 3B averages held-out SCC equally across the six datasets at each depth. The score grid reports genome-wide agreement, not chr1-only correlations.

D.2 Representation evaluation

The canonical cell order is fixed before embedding extraction and shared across all methods. Leiden clustering operates on each method's declared representation: 256 dimensions for Hi-Cell and PCA, 64 for Higashi and Va3DE, 10 for scHiCluster, 30 for LDA, and two for HiCRep/MDS. Scanpy constructs a 15-neighbor graph directly from these embeddings using Euclidean distance. The reported Leiden resolution is obtained by binary search within $[0, 3]$ to match the number of annotated classes; it is not chosen by maximizing NMI or ARI. The class count is therefore supplied to the clustering evaluation, while cell-level assignments are used to score the resulting partition. NMI uses arithmetic normalization of mutual information, and ARI uses the chance-adjusted agreement of cell pairs. UMAP is used only for visualization and never as classifier input.

For supervised readout, each dataset uses stratified five-fold cross-validation with shuffled outer folds. Hi-Cell's frozen embedding is standardized using training-fold statistics and passed to an MLP with one 128-unit hidden layer, GELU, dropout 0.2, and a linear class-output layer. Within an outer training partition, a stratified 15% validation split selects the training duration by validation cross-entropy. The standardizer is first fitted only to this inner training subset. Optimization uses AdamW with learning rate 10^{-3}, weight decay 10^{-4}, batch size 64, at most 200 epochs, patience

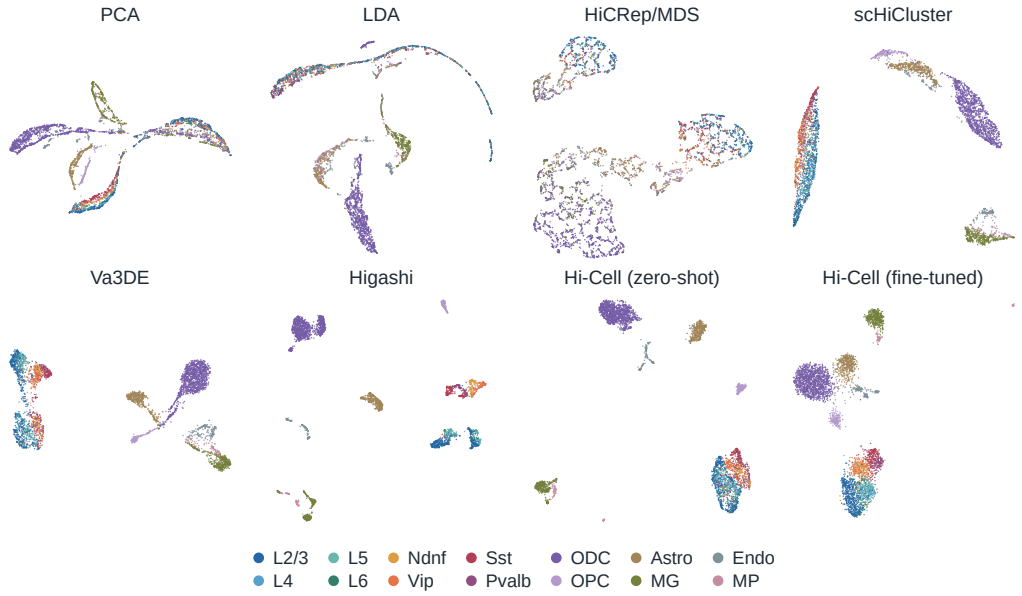


Figure S1: **Embedding comparison on the Lee2019 dataset.** Panels show UMAP projections of cell embeddings from the indicated methods. Points represent cells; colors indicate annotated cell types.

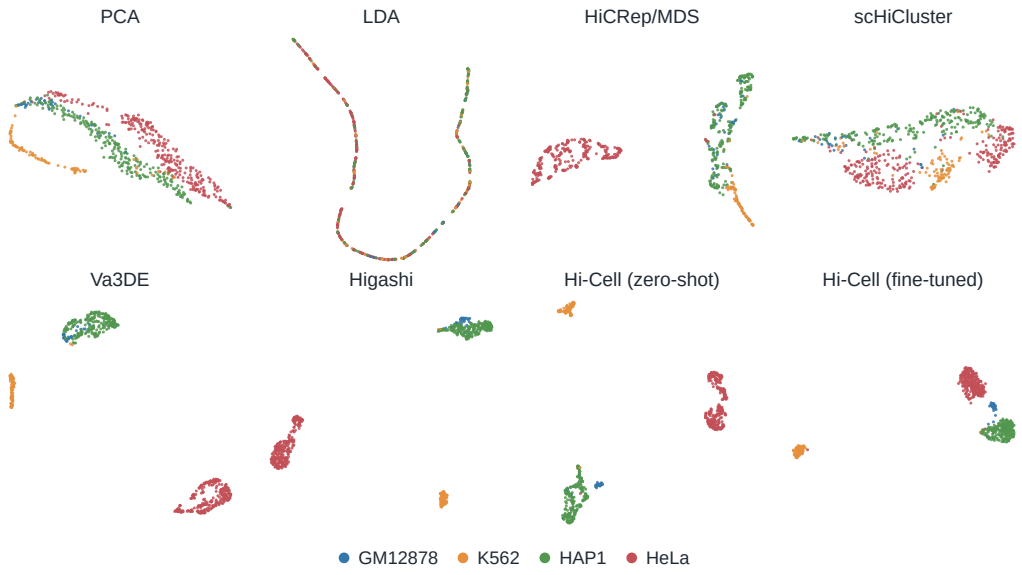


Figure S2: **Embedding comparison on the Ramani2017 dataset.** Panels show UMAP projections of cell embeddings from the indicated methods. Points represent cells; colors indicate cell-line identity.

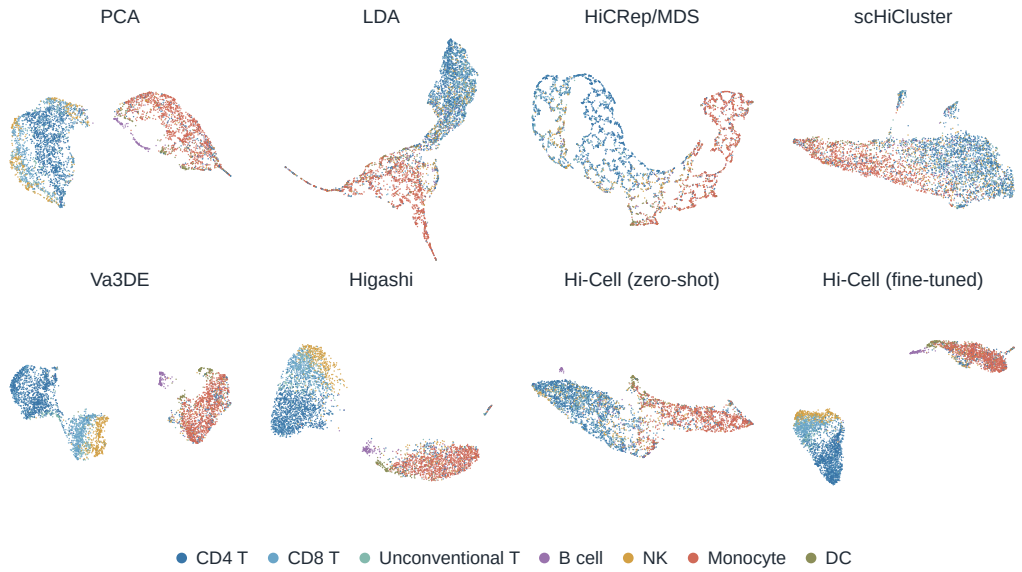


Figure S3: **Embedding comparison on the Chang2024 dataset.** Panels show UMAP projections of cell embeddings from the indicated methods. Points represent cells; colors indicate annotated cell types.

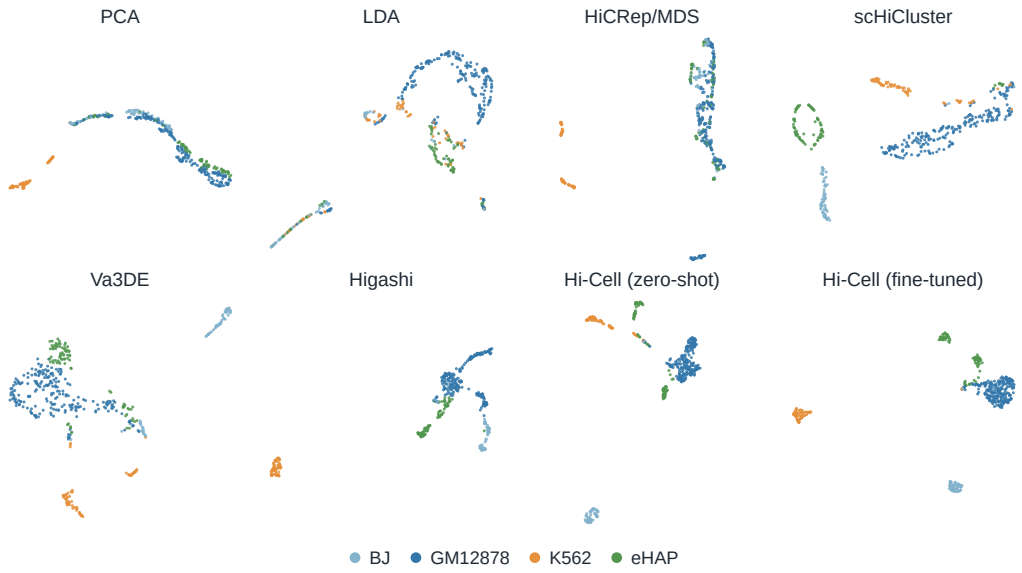


Figure S4: **Embedding comparison on the Wu2024 dataset.** Panels show UMAP projections of cell embeddings from the indicated methods. Points represent cells; colors indicate annotated cell types.

Table 8: **Balanced accuracy for cell-type annotation.** Balanced accuracy denotes mean class recall; Mean denotes the six-dataset average. Bold and underlined values indicate the highest and second-highest scores in each column, including ties at displayed precision.

Method	Ramani 2017	Lee 2019	Kim 2020	Chang 2024	Wu 2024	Zhou 2024	Mean
A. Balanced accuracy ↑							
Logistic regression	0.8427	0.7508	0.8383	<u>0.7459</u>	0.9790	0.5561	0.7855
Random forest	0.7480	0.4102	0.5897	0.5598	0.9481	0.4438	0.6166
scHiClassifier	0.8594	0.8654	0.7479	0.7364	0.8377	0.4466	0.7489
HI-CELL (sc-only)	0.9743	**0.8966**	0.9513	0.7293	<u>0.9886</u>	0.6258	0.8610
HI-CELL (zero-shot)	<u>0.9892</u>	<u>0.8862</u>	<u>0.9718</u>	0.7045	**0.9949**	**0.6815**	<u>0.8714</u>
HI-CELL (fine-tuned)	**0.9903**	0.8808	**0.9814**	**0.7528**	**0.9949**	<u>0.6460</u>	**0.8744**

25, and minimum improvement 10^{-5}. Cross-entropy uses inverse-frequency class weights $w_k = n/(Kn_k)$ computed from the corresponding training partition.

Table 8 reports balanced accuracy under this protocol. After selecting the epoch count, the standardizer and MLP are refitted from scratch on the entire outer training partition for that duration. The outer test fold is used only for prediction. Every cell receives exactly one out-of-fold prediction; concatenating these predictions gives the reported accuracy, balanced accuracy, and macro-F1. Balanced accuracy averages class recall, while macro-F1 averages class-specific F1 scores. A pooled metric is not the unweighted average of the five fold metrics. All five fold results and pooled scores are provided in Appendix F. Logistic regression, random forest, and scHiClassifier use the same outer test assignments. The encoder adaptation protocol and its relationship to these supervised folds are specified in Appendix C.

D.3 RECONSTRUCTION EVALUATION

Reconstruction evaluation retains cells with complete inputs and references. For each frozen target cell and depth, the input is sampled without replacement from its full contact multiset before any method runs, preserving integer multiplicity and never creating a contact absent from that cell. The three budgets are 10k, 25k, and 50k packed cis contacts. Cell order, chromosome coordinates, observed counts, full-depth reference, and evaluation support are fixed in the condition manifest and shared by all methods.

Held-out retrieval. Let x be the full-depth contact vector, o the observed input, and $h = x - o$ the held-out counts. Retrieval candidates are valid off-diagonal positions with $o_j = 0$; a candidate is positive if $h_j > 0$. Thus already observed positions do not receive retrieval credit. AUPRC is computed as average precision, $\sum_t (R_t - R_{t-1})P_t$, grouping tied scores at the same threshold. Precision@K uses K equal to the number of positive candidates for that cell and condition, not one fixed constant across datasets; a tie at the cutoff receives its expected positive fraction. Held-out Spearman correlation uses average ranks for ties and the same candidate set.

Distance-stratified correlation. Held-out SCC follows the stratum-adjusted correlation measure introduced by HiCRep (Yang et al., 2017), using h_j and predicted scores on that candidate set, stratified by exact genomic separation in 1 Mb bins. For eligible stratum d, let r_d be Pearson correlation, n_d the pair count, and $\sigma_{x,d}, \sigma_{y,d}$ the population standard deviations. The reported value is

$$\mathrm{SCC}(x, y) = \frac{\sum_d n_d \sigma_{x,d} \sigma_{y,d} r_d}{\sum_d n_d \sigma_{x,d} \sigma_{y,d}}. \tag{15}$$

Strata with fewer than three finite pairs or zero variance in either input are omitted. If no eligible stratum remains, the metric is undefined and retained as missing, not replaced by zero. This is an unsmoothed SCC calculation; it does not apply an additional HiCRep smoothing pipeline to model outputs.

Leave-one-out structural reference. For a cell i with annotated type y_i, the reference is $r_i = \sum_{k \neq i : y_k = y_i} x_k$ over the evaluated cells selected by the condition manifest, using their full-depth

maps. The evaluated cell is explicitly subtracted. SCC and Spearman compare the prediction with r_i over valid off-diagonal positions, including observed positions. Labels define this reference only and are not provided to the reconstruction model. Types without another reference cell cannot supply a leave-one-out score.

Distance-decay agreement. For each exact off-diagonal separation d, $P(d)$ is the mean score across its valid bin pairs. The curve is normalized to sum to one over the evaluated distances. Distance-decay error is $\sum_d |P_{\mathrm{LOO}}(d) - P_{\mathrm{pred}}(d)|$, comparing with the same leave-one-out pseudobulk reference used for structural correlation; lower values indicate better preservation.

All six datasets use the same preprocessing, three downsampling depths, and evaluator. Per-cell metric means are first computed within each dataset–depth condition; the main table then averages depths equally, and its Mean column averages datasets equally. Independent-bulk compartment validation computes chromosome PC1 from reconstructed observed/expected matrices, aligns the unavoidable eigenvector sign, and reports Pearson correlation, sign agreement, and sign macro-F1 against the external bulk profile.

A/B compartment analysis. Contact maps are aggregated by cell type before compartment analysis. The numerical pipeline applies square-root coverage normalization, distance-wise observed/expected transformation, row-correlation features, and one-component PCA. The observed/expected denominator is the mean of nonzero entries at each separation. For the native-input GM12878 and K562 comparison, methods and external bulk are independently analyzed on a common raw-pseudobulk reliable-bin mask; PC1 orientation uses available epigenetic calibration, or sign alignment to the bulk PC1 when calibration is absent. For the HeLa depth series, every method is projected into the same external HeLa-S3 bulk PCA basis, using its saved mean, component, and valid bins. The external HeLa reference accession is 4DNFIBM9QCFG. Genome-wide scores concatenate the retained autosomal bins; the displayed chromosome-1 tracks are standardized for visualization only. Missing bins remain gaps. The reference is used for this downstream structural comparison, not as an input to zero-shot contact reconstruction.

D.4 Pretraining and multiscale ablations

Figure 4A–C and Table 10 compare matched target-training settings. Scratch and Scratch-long use random initialization with approximately 1,200 and 4,800 updates, respectively; pretrained variants use approximately 1,200 updates. Panel C pairs Scratch-long with the pretrained model. S32, Fine, Mid, Coarse, and Full hierarchy retain patch sizes $\{32\}$, $\{8, 16\}$, $\{16, 32, 64\}$, $\{64, 128\}$, and $\{8, 16, 32, 64, 128\}$ bins, respectively, while retaining chromosome and cell pathways. Figure 4D compares frozen sc-only and final checkpoints, averaging SCC over the three contact budgets for each dataset. Reported cross-dataset means weight the six datasets equally.

D.5 Baseline implementations

We use the released baseline implementations and their native configurations, with the representation dimensions and task-specific settings summarized below; all outputs are aligned to the same canonical cells before evaluation.

PCA. PCA operates on flattened chromosome contact maps and retains 256 components.

LDA. Latent Dirichlet allocation (Kim et al., 2020) uses binarized near-diagonal contacts and the implementation's 30-topic configuration.

HiCRep/MDS. HiCRep-derived cell similarities (Yang et al., 2017) are converted to a two-dimensional MDS representation using the baseline pipeline.

scHiCluster. The native convolution and random-walk pipeline (Zhou et al., 2019) supplies both imputed contact maps and its 10-component cell representation.

Table 9: **Contact retrieval, rank correlation, and distance-decay metrics at 1 Mb.** Panels show precision@K, held-out and leave-one-out (LOO) Spearman correlations, and $P(s)$ L1 error. K denotes the number of positive held-out positions; $P(s)$ denotes the contact-distance distribution. Mean denotes the six-dataset average. Arrows indicate the metric direction; bold and underlining mark first and second ranks, including ties at displayed precision. – denotes undefined correlations.

Method	Ramani 2017	Lee 2019	Kim 2020	Chang 2024	Wu 2024	Zhou 2024	Mean
A. Precision@K ↑							
Observed input	0.0164	0.0912	0.0644	0.0041	0.1634	0.0700	0.0683
scHiCluster	0.1851	**0.3950**	0.2817	0.0263	0.4701	0.3594	0.2863
Higashi-0	0.2096	**0.3950**	0.2926	0.0153	**0.4761**	**0.3770**	0.2943
Higashi-neighbor	0.2321	0.3628	0.3199	0.0210	0.4682	0.3616	0.2943
HI-CELL (sc-only)	0.2768	0.3445	0.3369	0.0404	0.4601	0.3449	0.3006
HI-CELL (zero-shot)	**0.2853**	0.3402	**0.3482**	**0.0802**	0.4678	0.3496	**0.3119**
B. Held-out Spearman ↑							
Observed input	–	–	–	–	–	–	–
scHiCluster	0.1431	**0.2975**	0.2301	0.0446	0.3844	**0.2899**	**0.2316**
Higashi-0	0.1368	0.2875	0.2258	0.0225	0.3883	0.2871	0.2247
Higashi-neighbor	0.1363	0.2699	0.2414	0.0306	0.3841	0.2749	0.2229
HI-CELL (sc-only)	0.1442	0.2605	0.2509	0.0491	0.3858	0.2674	0.2263
HI-CELL (zero-shot)	**0.1480**	0.2498	**0.2617**	**0.0582**	**0.3891**	0.2621	0.2281
C. LOO pseudobulk Spearman ↑							
Observed input	0.2125	0.2199	0.2390	0.0251	0.2058	0.2164	0.1865
scHiCluster	0.5014	0.6015	0.6134	0.0755	0.6454	0.6270	0.5107
Higashi-0	0.5395	0.6414	0.6299	0.0944	0.7309	0.7172	0.5589
Higashi-neighbor	0.5818	0.6999	0.6921	0.1448	0.7672	0.7634	0.6082
HI-CELL (sc-only)	0.6014	0.7905	0.7167	0.1824	0.8179	0.8211	0.6550
HI-CELL (zero-shot)	**0.6629**	**0.8767**	**0.7755**	**0.2252**	**0.8602**	**0.9004**	**0.7168**
D. $P(s)$ L1 error ↓							
Observed input	0.3298	0.3052	0.3229	0.9035	0.4077	0.2757	0.4241
scHiCluster	0.5152	0.3389	0.4617	0.5759	0.4095	0.3285	0.4383
Higashi-0	0.5829	0.4074	0.5765	0.8535	0.4404	0.3564	0.5362
Higashi-neighbor	0.5744	0.4031	0.5165	0.5692	0.4396	0.3266	0.4716
HI-CELL (sc-only)	0.1936	0.1365	0.1726	0.8108	0.2157	0.1444	0.2789
HI-CELL (zero-shot)	**0.1869**	**0.1002**	**0.1522**	**0.4989**	**0.1891**	**0.1031**	**0.2051**

Table 10: **Pretraining and patch-scale ablations across six datasets.** Columns show mean normalized mutual information (NMI), adjusted Rand index (ARI), annotation accuracy (Acc.), and macro-F1. Braces list patch sizes in bins. Bold indicates the highest value in each column.

Variant	NMI	ARI	Acc.	Macro-F1
Scratch	.0448	.0216	.6245	.5425
Scratch-long	.4047	.3391	.7711	.6929
Full hierarchy	.7225	**.7195**	**.9004**	**.8685**
Single scale 32	.6965	.6567	.8925	.8524
Fine $\{8, 16\}$	**.7247**	.7146	.8996	.8667
Mid $\{16, 32, 64\}$	.7044	.6712	.8990	.8659
Coarse $\{64, 128\}$	.6876	.6372	.8934	.8566

Va3DE. The released variational embedding pipeline (Plummer et al., 2025) supplies a 64-dimensional cell representation.

Higashi. The native pipeline (Zhang et al., 2022) supplies 64-dimensional embeddings and contact predictions without neighbor aggregation (Higashi-0) or with five-neighbor aggregation (Higashi-neighbor).

Annotation baselines. Logistic regression, random forest, and scHiClassifier follow the feature constructions and implementations of Zhou & Wu (2025), adapted to the same canonical cells and five outer test folds, with any hyperparameter selection restricted to the outer training partitions.

E Corpus Coverage and Data Provenance

Human-Hi-C-Corpus combines public single-cell and bulk Hi-C datasets. Tissue coverage is calculated from the training inputs, using saved sampling configurations and source metadata. Cell identifiers, quality-control records, genomic coordinates, and source bulk-map identifiers link each training input to its original dataset.

The real-cell sources are the adult-brain atlas (Tian et al., 2023), developing-brain atlas (Heffel et al., 2024), subcutaneous-adipose atlas (Chen et al., 2025b), cardiomyocyte-differentiation study (Grant et al., 2026), retinal atlas (Yuan et al., 2026), and human-body atlas (Zhou et al., 2026). Table 11 summarizes the tissues and cell types included in pretraining. Bulk accession identifiers, source-publication labels, biological contexts, and training/validation partitions are retained in the accompanying data-provenance manifest; records obtained through 4DN are attributed to the Data Portal (Reiff et al., 2022).

Table 11: **Real scHi-C pretraining datasets and their biological coverage.** Rows list source datasets and their tissues, anatomical regions, cell types, or developmental stages.

Source	Tissue and anatomical contexts
Tian2023	17 brain regions; 29 major cell types and 156 annotated subtypes
Heffel2024	Cortex and hippocampus; second/third trimester, infancy, and adulthood
Chen2025	Human subcutaneous adipose tissue
Grant2025	In-vitro atrial and ventricular cardiomyocyte differentiation
Yuan2025	Retina, macula, and retinal pigment epithelium
Zhou2025	15 tissues, including breast and heart

Tissue coverage. The 17 tissues are adrenal gland, blood, brain, breast, colon, esophagus, eye, heart, lung, pancreas, peripheral nerve, placenta, skeletal muscle, skin, small intestine, stomach, and subcutaneous adipose tissue. Source mappings and tissue annotations are provided in the accompanying data-provenance records.

Bulk Hi-C conditions. The eight bulk stem-cell/differentiation states are undifferentiated pluripotent cells, mesoderm, cardiogenic mesoderm, cardiac muscle myoblasts, primitive cardiac myocytes, cardiac muscle cells, definitive endoderm, and foregut endoderm. The bulk training partition additionally spans 21 time points from 0.5 to 12 hours after release from prometaphase arrest and perturbation contexts involving RAD21, NIPBL, WAPL, SCC4, and TBX5.

Separation of training and evaluation information. Single-cell pretraining, target-dataset adaptation, classifier fitting, and reference-based evaluation have separate roles. The external datasets do not provide single-cell pretraining contact profiles. Their labels do not update the pretrained backbone or select adaptation checkpoints. Aggregate QC calibration of pseudo-single-cell depth is disclosed in Appendix B. Class labels enter the supervised readout, clustering class-count matching and scoring, and construction of evaluation-only leave-one-out references.

F Complete Five-Fold Annotation Results

Tables 12–14 report every outer test fold for the six annotation methods in the main comparison. Dataset order follows publication year. Pooled scores are computed from concatenated out-of-fold predictions and reproduce the main tables; they are not recomputed by averaging the displayed fold scores. The five columns are test folds, not independent pretraining runs.

Table 12: **Accuracy in all five outer folds.** Pooled denotes the score across all held-out folds.

Method	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Pooled
Ramani2017						
Logistic regression	0.9758	0.9593	0.9756	0.9593	0.9593	0.9659
Random forest	0.9435	0.9512	0.9512	0.9350	0.9431	0.9448
scHiClassifier	0.9758	0.9675	0.9756	0.9756	0.9593	0.9708
HI-CELL (sc-only)	0.9839	0.9756	0.9837	0.9919	0.9837	0.9838
HI-CELL (zero-shot)	0.9919	1.0000	0.9919	0.9919	0.9919	0.9935
HI-CELL (fine-tuned)	1.0000	1.0000	0.9919	1.0000	0.9837	0.9951
Lee2019						
Logistic regression	0.8514	0.8561	0.8715	0.8937	0.8666	0.8679
Random forest	0.6922	0.6910	0.7123	0.7096	0.7096	0.7029
scHiClassifier	0.9304	0.9198	0.9127	0.9398	0.9256	0.9257
HI-CELL (sc-only)	0.9434	0.9434	0.9328	0.9433	0.9492	0.9424
HI-CELL (zero-shot)	0.9316	0.9304	0.9281	0.9280	0.9315	0.9299
HI-CELL (fine-tuned)	0.9375	0.9340	0.9151	0.9351	0.9268	0.9297
Kim2020						
Logistic regression	0.9810	0.9829	0.9791	0.9823	0.9810	0.9813
Random forest	0.8550	0.8619	0.8486	0.8626	0.8612	0.8579
scHiClassifier	0.9405	0.9177	0.9405	0.1305	0.9328	0.7724
HI-CELL (sc-only)	0.9759	0.9924	0.9759	0.9835	0.9816	0.9819
HI-CELL (zero-shot)	0.9956	0.9962	0.9911	0.9930	0.9918	0.9935
HI-CELL (fine-tuned)	0.9924	0.9981	0.9956	0.9987	0.9924	0.9954
Chang2024						
Logistic regression	0.8280	0.8535	0.8456	0.8467	0.8111	0.8370
Random forest	0.7481	0.7802	0.7633	0.7878	0.7700	0.7699
scHiClassifier	0.8513	0.8690	0.8411	0.8456	0.6944	0.8203
HI-CELL (sc-only)	0.7780	0.7603	0.7700	0.7567	0.7622	0.7654
HI-CELL (zero-shot)	0.7281	0.6926	0.7400	0.7133	0.7156	0.7179
HI-CELL (fine-tuned)	0.8036	0.8058	0.7900	0.7967	0.7722	0.7936
Wu2024						
Logistic regression	0.9767	1.0000	0.9884	0.9767	0.9765	0.9837
Random forest	0.9651	0.9884	0.9651	0.9651	0.9176	0.9604
scHiClassifier	0.9767	1.0000	0.1512	0.9884	0.9882	0.8205
HI-CELL (sc-only)	0.9884	0.9884	1.0000	0.9767	0.9882	0.9883
HI-CELL (zero-shot)	0.9884	1.0000	1.0000	0.9884	1.0000	0.9953
HI-CELL (fine-tuned)	0.9884	1.0000	1.0000	0.9884	1.0000	0.9953
Zhou2024						
Logistic regression	0.6726	0.6845	0.6886	0.6467	0.6647	0.6714
Random forest	0.6012	0.6250	0.5928	0.6287	0.5808	0.6057
scHiClassifier	0.2262	0.7440	0.7066	0.2156	0.6587	0.5102
HI-CELL (sc-only)	0.6369	0.6786	0.7126	0.6647	0.6647	0.6714
HI-CELL (zero-shot)	0.6905	0.7024	0.7126	0.7066	0.7126	0.7049
HI-CELL (fine-tuned)	0.6845	0.6845	0.7425	0.6766	0.6766	0.6930

Table 13: **Balanced accuracy in all five outer folds.** Pooled denotes the score across all held-out folds.

Method	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Pooled
Ramani2017						
Logistic regression	0.8929	0.7917	0.8750	0.8280	0.8214	0.8427
Random forest	0.7500	0.7500	0.7500	0.7400	0.7500	0.7480
scHiClassifier	0.8929	0.8333	0.8750	0.8750	0.8214	0.8594
HI-CELL (sc-only)	0.9891	0.9744	0.9900	0.9953	0.9286	0.9743
HI-CELL (zero-shot)	0.9946	1.0000	0.9953	0.9953	0.9643	0.9892
HI-CELL (fine-tuned)	1.0000	1.0000	0.9953	1.0000	0.9596	0.9903
Lee2019						
Logistic regression	0.7226	0.7438	0.7510	0.8024	0.7333	0.7508
Random forest	0.3951	0.3922	0.4253	0.4165	0.4222	0.4102
scHiClassifier	0.8846	0.8431	0.8513	0.8917	0.8561	0.8654
HI-CELL (sc-only)	0.8972	0.8886	0.8950	0.8998	0.9019	0.8966
HI-CELL (zero-shot)	0.8969	0.8830	0.8905	0.8930	0.8674	0.8862
HI-CELL (fine-tuned)	0.8960	0.8807	0.8693	0.8934	0.8637	0.8808
Kim2020						
Logistic regression	0.8027	0.8258	0.8630	0.8214	0.8804	0.8383
Random forest	0.5868	0.5926	0.5831	0.5957	0.5901	0.5897
scHiClassifier	0.8881	0.8325	0.9035	0.2308	0.8942	0.7479
HI-CELL (sc-only)	0.9423	0.9538	0.9437	0.9670	0.9481	0.9513
HI-CELL (zero-shot)	0.9766	0.9966	0.9920	0.9415	0.9556	0.9718
HI-CELL (fine-tuned)	0.9742	0.9988	0.9775	0.9985	0.9564	0.9814
Chang2024						
Logistic regression	0.7455	0.7623	0.7455	0.7637	0.7125	0.7459
Random forest	0.5376	0.5518	0.5474	0.5914	0.5707	0.5598
scHiClassifier	0.7662	0.7914	0.7565	0.7825	0.5837	0.7364
HI-CELL (sc-only)	0.7352	0.7335	0.7066	0.7398	0.7315	0.7293
HI-CELL (zero-shot)	0.6937	0.7132	0.7093	0.7015	0.7040	0.7045
HI-CELL (fine-tuned)	0.7480	0.7850	0.7425	0.7563	0.7317	0.7528
Wu2024						
Logistic regression	0.9651	1.0000	0.9853	0.9796	0.9645	0.9790
Random forest	0.9495	0.9853	0.9514	0.9588	0.8937	0.9481
scHiClassifier	0.9651	1.0000	0.2500	0.9853	0.9792	0.8377
HI-CELL (sc-only)	0.9808	0.9943	1.0000	0.9735	0.9942	0.9886
HI-CELL (zero-shot)	0.9808	1.0000	1.0000	0.9943	1.0000	0.9949
HI-CELL (fine-tuned)	0.9808	1.0000	1.0000	0.9943	1.0000	0.9949
Zhou2024						
Logistic regression	0.5757	0.5826	0.5678	0.5165	0.5387	0.5561
Random forest	0.4381	0.4623	0.4292	0.4667	0.4222	0.4438
scHiClassifier	0.1817	0.6967	0.6154	0.1667	0.5741	0.4466
HI-CELL (sc-only)	0.5804	0.6439	0.7030	0.6072	0.5939	0.6258
HI-CELL (zero-shot)	0.6955	0.7390	0.6906	0.6558	0.6264	0.6815
HI-CELL (fine-tuned)	0.6408	0.6337	0.7080	0.6336	0.6141	0.6460

Table 14: **Macro-F1 in all five outer folds.** Pooled denotes the score across all held-out folds.

Method	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Pooled
Ramani2017						
Logistic regression	0.9239	0.8088	0.9089	0.8482	0.8484	0.8725
Random forest	0.7323	0.7350	0.7350	0.7253	0.7323	0.7320
scHiClassifier	0.9239	0.8648	0.9089	0.9089	0.8482	0.8929
HI-CELL (sc-only)	0.9850	0.9625	0.9900	0.9950	0.9485	0.9765
HI-CELL (zero-shot)	0.9806	1.0000	0.9784	0.9950	0.9781	0.9859
HI-CELL (fine-tuned)	1.0000	1.0000	0.9905	1.0000	0.9686	0.9916
Lee2019						
Logistic regression	0.7348	0.7543	0.7593	0.8123	0.7513	0.7642
Random forest	0.3813	0.3771	0.4086	0.4067	0.4075	0.3975
scHiClassifier	0.8836	0.8502	0.8498	0.8958	0.8649	0.8700
HI-CELL (sc-only)	0.8973	0.8901	0.8791	0.8951	0.9024	0.8921
HI-CELL (zero-shot)	0.8831	0.8784	0.8777	0.8763	0.8762	0.8782
HI-CELL (fine-tuned)	0.8894	0.8816	0.8530	0.8849	0.8624	0.8740
Kim2020						
Logistic regression	0.8178	0.8506	0.8968	0.8461	0.9158	0.8692
Random forest	0.5650	0.5710	0.5613	0.5785	0.5651	0.5682
scHiClassifier	0.8605	0.8622	0.8481	0.0966	0.8458	0.7102
HI-CELL (sc-only)	0.9317	0.9711	0.9349	0.9685	0.9542	0.9518
HI-CELL (zero-shot)	0.9860	0.9969	0.9536	0.9629	0.9549	0.9700
HI-CELL (fine-tuned)	0.9737	0.9987	0.9856	0.9988	0.9723	0.9863
Chang2024						
Logistic regression	0.7756	0.7903	0.7827	0.7872	0.7275	0.7729
Random forest	0.5816	0.5908	0.5898	0.6339	0.6103	0.6019
scHiClassifier	0.7940	0.8153	0.7825	0.8036	0.5801	0.7506
HI-CELL (sc-only)	0.7122	0.7011	0.6895	0.6906	0.6923	0.6964
HI-CELL (zero-shot)	0.6541	0.6502	0.6757	0.6595	0.6409	0.6551
HI-CELL (fine-tuned)	0.7441	0.7524	0.7142	0.7372	0.7065	0.7297
Wu2024						
Logistic regression	0.9699	1.0000	0.9832	0.9796	0.9687	0.9803
Random forest	0.9585	0.9896	0.9596	0.9632	0.9026	0.9553
scHiClassifier	0.9699	1.0000	0.0657	0.9896	0.9863	0.8162
HI-CELL (sc-only)	0.9872	0.9879	1.0000	0.9763	0.9878	0.9877
HI-CELL (zero-shot)	0.9872	1.0000	1.0000	0.9900	1.0000	0.9954
HI-CELL (fine-tuned)	0.9872	1.0000	1.0000	0.9900	1.0000	0.9954
Zhou2024						
Logistic regression	0.5855	0.5884	0.5868	0.5182	0.5498	0.5688
Random forest	0.4069	0.4289	0.3991	0.4332	0.3862	0.4122
scHiClassifier	0.1151	0.7143	0.6383	0.0591	0.5751	0.4537
HI-CELL (sc-only)	0.5709	0.6264	0.6750	0.5882	0.6039	0.6126
HI-CELL (zero-shot)	0.6495	0.6679	0.6769	0.6366	0.6261	0.6568
HI-CELL (fine-tuned)	0.6249	0.6232	0.7034	0.6086	0.6154	0.6340

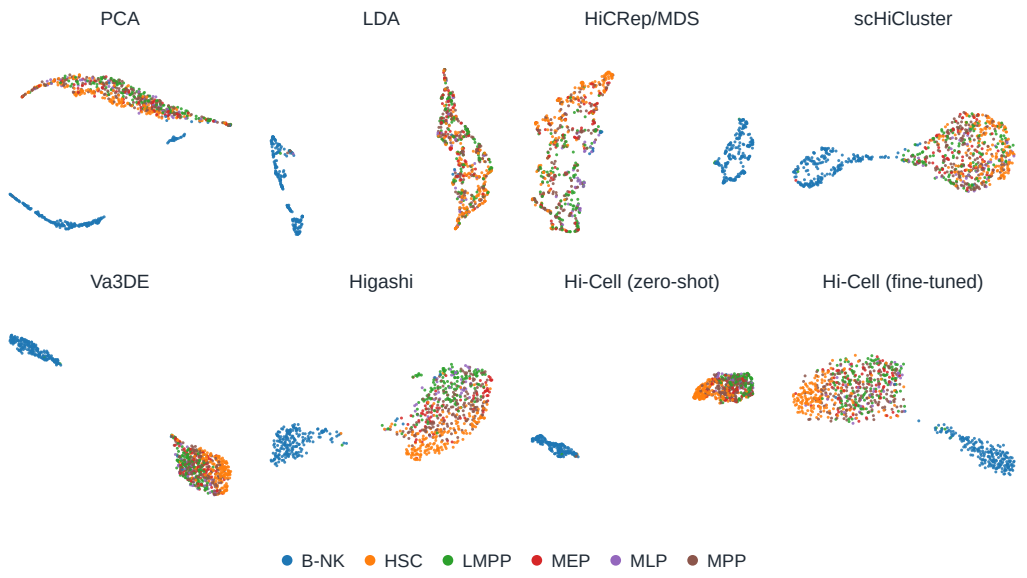


Figure S5: **Embedding comparison on the Zhou2024 dataset.** Panels show UMAP projections of cell embeddings from the indicated methods. Points represent cells; colors indicate annotated cell types.