ARTICLES
https://doi.org/10.1038/s42256-022-00518-z

nature machine intelligence

Contrastive learning enables rapid mapping to multimodal single-cell atlas of multimillion scale

Meng Yang[1,2,5]✉, Yueyuxiao Yang[1,5], Chenxi Xie[1,5], Ming Ni[3], Jian Liu[1], Huanming Yang[4], Feng Mu[1]✉ and Jian Wang[1,4]✉

Single-cell datasets continue to grow in size, posing computational challenges for dealing with expanded scale, extended modality and inevitable batch effects. Deep learning-based approaches have recently emerged to address these points by deriving nonlinear cell embeddings. Here we present contrastive learning of cell representations, Concerto, which leverages a self-supervised distillation framework to model multimodal single-cell atlases. Simply by discriminating each cell from the others, Concerto can be adapted to various downstream tasks such as automatic cell type classification, data integration and especially reference mapping. Unlike current mainstream packages, Concerto's contrastive setting well supports operating on all genes to preserve biological variations. Concerto can flexibly generalize to multiomics to obtain unified cell representations. Benchmarking on both simulated and real datasets, Concerto substantially outperforms competing methods. By mapping to a comprehensive reference, Concerto recapitulates differential immune responses and discovers disease-specific cell states in patients with COVID-19. Concerto is easily parallelizable and efficiently scalable to build a 10-million-cell reference within 1.5 h and query 10,000 cells within 8 s. Overall, Concerto will facilitate biomedical research by enabling iteratively constructing single-cell reference atlases and rapidly mapping novel dataset against them to transfer relevant cell annotations.

Recent single-cell multiomics tools are revolutionizing tissue characterization at unprecedented resolutions. The large-scale Human Cell Atlas[1] and Tabula Muris Atlas[2] are approaching the multimillion scale. Computational pipelines such as Seurat[3], SCANPY[4] and Pegasus[5] have been developed and benchmarked[6–8]. Single-cell analysis has some unique features. First, the overabundance of zero counts can either be due to technical drop-outs or biological signals. Debate continues regarding the underlying distribution[9]. Mainstream packages rely on feature selection (highly variable genes, HVGs) and linear dimension reduction (principal component analysis, PCA[10]) to extract major variations, which might cause information loss. The deep learning approach offers a promising solution to model nonlinear relationships among all genes. Variational autoencoders (VAEs) apply encoder–decoder structures with reconstruction functions to learn low-dimensional cell embeddings (scVI[11], scETM[12]). However, forcing the model to reconstruct ambiguous zeroes deserves further discussion. Second, batch effects widely exist across technologies, conditions and donors. Disentangling biological signals from confounding effects is important for data integration. Seurat v.3 (ref. [13]) identifies anchor cell pairs across batches using mutual nearest neighbours[14,15], which only allows integration of two batches at a time and incurs exponentially boosted memory consumption when processing more cells. Harmony[16] iteratively uses fuzzy clustering and linear correction, whereas trVAE[17] leverages a conditional VAE to correct batch effects. The ideal method should be scaled to a million-scale, integrate multiple batches simultaneously and avoid mixing non-overlapping populations. Finally, query-to-reference mapping has become popular to enable quick interpretation of newly generated datasets without laborious de novo clustering or manual annotation[18]. Unlike rigid supervised classification, we consider query-to-reference mapping as an unsupervised transfer learning problem to derive voting-based annotations based on learned query embeddings. Seurat v.4 uses[19] supervised PCA on mutual nearest neighbours to transfer reference annotations. Symphony[20] uses a mixture modelling framework to localize queries onto the stable reference. ScArches[18] uses a conditional autoencoder to map query cells through fine-tuning.

Contrastive learning has recently achieved great success in computer vision domains such as SimCLR[21] and MoCo[22]. This type of method defines a pretext task for unlabelled images and conducts self-supervised learning by minimizing contrastive loss between augmented views in hypersphere space[23]. Learned embeddings can be used for image classification through fine-tuning, considerably outperforming previous approaches[24]. Inspired by contrastive learning's superiority in modelling unlabelled data, we anticipate that high-quality representations can be obtained simply by discriminating between each cell in a self-supervised manner. Distillation schemes have also been used to transfer knowledge between asymmetric neural networks (that is, teacher–student networks), evolving from model compression[25] and online co-distillation[26] in a supervised setting, to self-training in a semi-supervised setting (for example, noisy student[27]), to self-supervised distillation for better representations (for example, SEED[28], DINO[29]). Exploiting a distillation-like scheme to share knowledge between augmented views provides a concise solution to generate self-consistent but unique embeddings in a typical contrastive learning framework.

Here we propose a self-distillation contrastive learning framework for single-cell analysis, Concerto. Through a comprehensive benchmark on real and simulated datasets, learned embeddings can be fine-tuned for various downstream needs, covering automatic cell type classification, clustering, data integration for batch-effect correction, and query-to-reference mapping. Concerto can flexibly handle multiomics datasets and achieve superior performance

[1]MGI, BGI-Shenzhen, Shenzhen, China. [2]Department of Biology, University of Copenhagen, Copenhagen, Denmark. [3]MGI-QingDao, BGI-Shenzhen, Qingdao, China. [4]BGI-Shenzhen, Shenzhen, China. [5]These authors contributed equally: Meng Yang, Yueyuxiao Yang and Chenxi Xie.
✉e-mail: yangmeng1@mgi-tech.com; mufeng@mgi-tech.com; wangjian@genomics.cn

to competing methods in each task. We also show that Concerto's attention weights offer model interpretability by extracting molecular signatures at single-cell resolution. Moreover, we leverage Concerto to query a COVID-19 immune cell dataset against an integrated reference atlas containing both healthy and infected samples, recapitulating several differential immune features among patients with diverse disease statuses. Concerto is a robust, accurate, scalable representation learning framework for single-cell multimodal analysis at the 10-million-cell scale.

Results

Overview of Concerto architecture. Concerto leverages a self-distillation contrastive learning framework configured as an asymmetric teacher–student architecture (Fig. 1a and Methods). The asymmetric design injects imbalanced model complexity, where a larger teacher network aggregates gene embeddings into cell embeddings via an attention mechanism[30,31] and a smaller student network simply transforms discrete inputs into cell embeddings using a dense operation. Representational knowledge is transferred in between by self-distillation. By defining an instance discrimination pretext task for each unlabelled cell, Concerto learns semantic-invariant embeddings by maximizing the agreement between each cell's teacher and student views. A random dropout mask[32] is added right before the output layer to generate minimal data augmentations at the model level, motivated by SimCSE's[33] sentence-processing scheme. A domain-specific batch normalization layer is added to correct for batch effects[34]. When processing a multiomics dataset, simple element-wise summation for each modality can generate unified cell embeddings (Fig. 1b and Methods). By projecting onto unit hypersphere space[23], Concerto discriminates between cells by pulling together teacher–student views of the same cell as positive pairs while pushing apart other cells within a batch. Learned embeddings can be fine-tuned for various downstream tasks, including automatic cell type classification, clustering, data integration for batch-effect correction, and query-to-reference mapping (Fig. 1c; see the Methods for details). The rationale for choosing different components is discussed in the Supplementary Notes.

Contrastively learned embeddings notably boost the performance of automatic cell classification via fine-tuning and support novel cell type discovery across tissues. To demonstrate that contrastively learned embeddings satisfy rigid cell classification, we use existing annotations as training labels to implement supervised fine-tuning on Concerto. First, we use a classical human peripheral blood mononuclear cell dataset (PBMC45k, $n = 31{,}021$, seven protocols)[35] to compare different classifiers (Methods), including likelihood-based SciBet[36], neural network-based Cell BLAST[37], correlation-based SingleR[38], support vector machine-based Moana[39], and a meta-learning approach, MARS[40]. Concerto is a two-step approach (pretraining and fine-tuning) whereas others are trained end-to-end. We also implement an end-to-end version of Concerto (Concerto-E2E) by discarding the contrastive loss and training it in a fully supervised manner. For intra-dataset evaluation, we apply fivefold cross-validation within each batch ($n = 9$) and evaluate the median F1-score across all cell types. Concerto achieves the highest score (0.926) with the most stable performance across each fold (Fig. 2a), whereas Concerto-E2E obtains a lower score (0.867), demonstrating the utility brought by pretraining (see Supplementary Fig. 1 for details). For inter-dataset assessment, we use one protocol as the test set and the other protocols as the training set (bootstrapping five times). Concerto substantially outperforms other methods on almost all train–test splits (Fig. 2b). When the Seq-well dataset is held-out, all methods report a decline in performance, probably because the microwell-based protocol markedly contrasts with droplet-based methods, posing greater challenges to model transferability (see Supplementary Fig. 2 for details).

A good classifier should label none-of-the-above (NOTA) cells as a rejection option if the test set contains cell types that do not exist in the training samples. We download the PBMC CITE-seq dataset (PBMC160k, $n = 161{,}764$ cells, RNA-only in the NOTA study) annotated at three levels and remove different T cell granularities from the training set to evaluate the NOTA setting (Methods). Figure 2d shows that Concerto can clearly separate the confidence curves of the validation and test sets for level-1 and level-2 masking. Even for the most challenging level-3 scenario, Concerto obtains a bimodal curve with partial overlap with the validation curve; nevertheless, SciBet misassigns CD4 Mem T cells as other types (see Supplementary Fig. 4–6 for details). We use Splatter[41] to conduct robustness analysis on simulated datasets (Methods). Concerto obtains the highest accuracy (ACC) value when decreasing intensities of differential expression at a fixed dropout rate or increasing dropout rate at a fixed expression variance (Supplementary Fig. 8).

To benchmark on finer-grained classification, we combined the Thymus scRNA-seq atlas[42] ($n = 107{,}969$ cells) with PBMC45k to construct a multihierarchical immune cell dataset. Incorporating a high-resolution thymus dataset poses a greater challenge in distinguishing subtle state discrepancies along the T cell development trajectory. Concerto still reaches the highest median F1-score of 0.830 (mean value for fivefold cross-validation), substantially outperforming SingleR (0.705) and SciBet (0.667) (Fig. 2c). Concerto can well discriminate between different developmental stages, including double-negative T cells, double-positive T cells and single-positive T cells. We also use the heterogenous *Tabula Muris Senis* (TMS) atlas ($n = 101{,}045$ cells, 23 mouse tissues) to train a tissue-wise classifier (intra-tissue prediction, fivefold cross-validation). Concerto outperforms SciBet on all tissues by a large margin (Fig. 2f), achieving the top mean ACC for the bladder (0.999), brain myeloid (0.999) and mammary gland (0.996). The largest absolute gain over SciBet is for the tongue (+7.85%), large intestine (+7.69%) and brown adipose tissue (+7.26%) (see Supplementary Fig. 3 for details).

For cross-tissue annotations, we adopt a similar experimental design to MARS[40] by leaving one tissue out as an unannotated test set and training Concerto on all of the other tissues (TMS dataset). By adding a domain adaptation module[43] (Methods), Concerto achieves a superior adjusted Rand index (ARI) to MARS on 22 hold-out tissues, ranging from the largest absolute gain of ARI for the spleen (+89.4%) to the smallest for the bladder (+0.613%) (mean value, bootstrapping three times; Fig. 2g). The hold-out tissue often contains several cell types that do not exist in training tissues. Similar to MARS, Concerto effectively transfers knowledge to discover novel cell types across tissues (see the spleen and brain non-myeloid results in Supplementary Fig. 7). In particular, when the limb muscle is held-out, Concerto places functionally similar cell types from other tissues closer to the limb muscle's six major annotations (Fig. 2h). The Sankey plot (Fig. 2i) shows that general B cells, T cells, endothelial cells and macrophages from other tissues are correctly transferred to the limb muscle. Skeletal muscle satellite cells and mesenchymal stem cells from the limb muscle correctly map to their counterparts in other muscles and adipose tissues, whereas MARS erroneously uses T cells to annotate some satellite cells in the limb muscle.

To assess Concerto's capability to process multiomics data, we use PBMC160k[19] to train Concerto in three settings: with RNA, with protein, or with both, as input. Concerto achieves a median F1-score of 0.805, 0.770 and 0.819, respectively (mean value of fivefold cross-validation, intra-dataset prediction), implying that unifying multimodalities enable more accurate classification (Fig. 2e). Concerto outperforms Azimuth in all cases, obtaining an absolute improvement of 4.8% when using dual-modality as input.

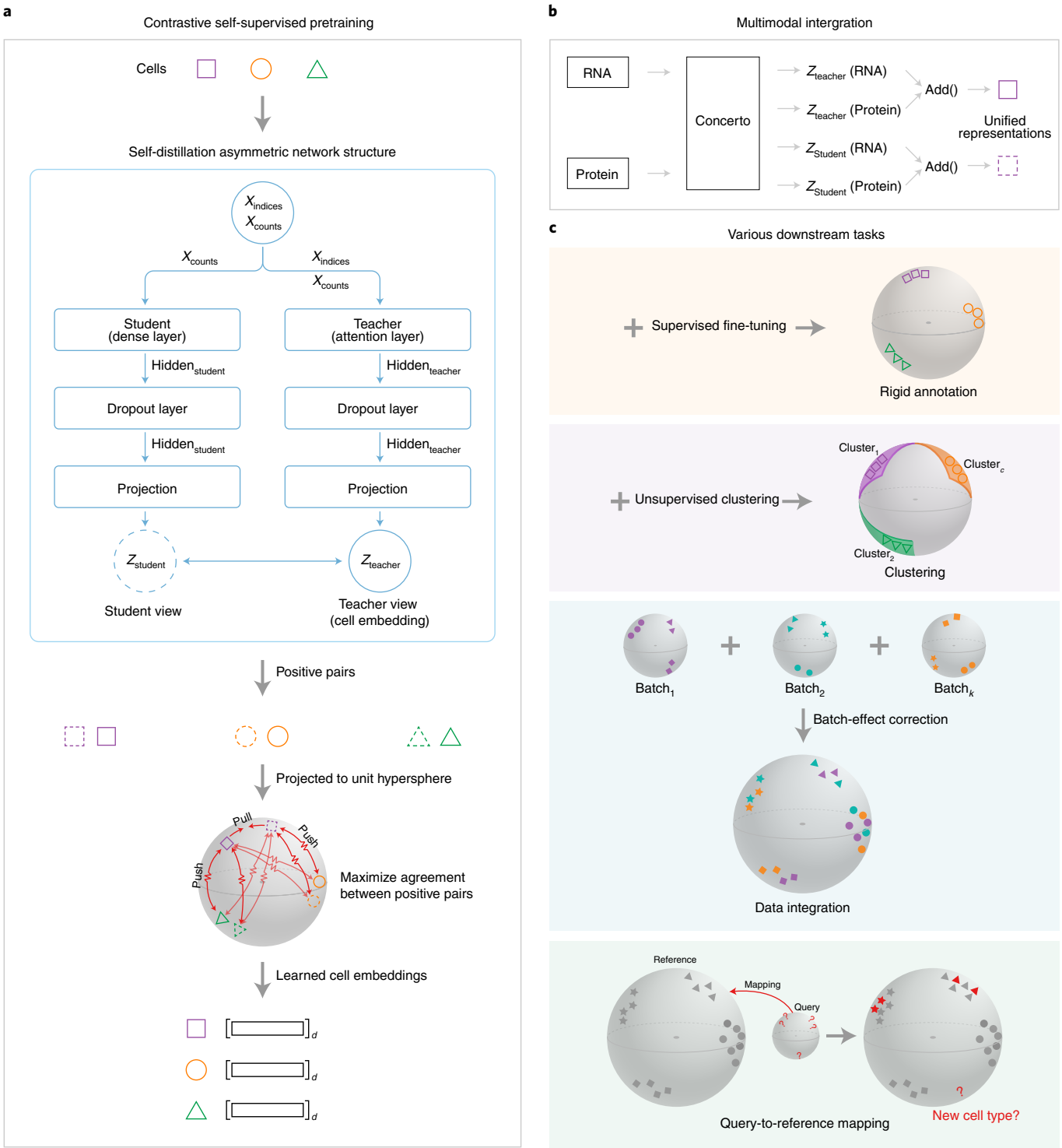


Fig. 1 | Overview of Concerto architecture. a, Each cell obtains two views through an asymmetric teacher–student network structure (blue box, forming a self-distillation scheme, where the solid and dashed lines represent the output from the teacher and student views, respectively. Each cell's two views are positive pairs to be pulled together while views from other cells are pushed away, together forming an optimization objective as contrastive leaning. **b**, When using a multiomics dataset as input, Concerto simply implements element-wise summation of the outputs from different modalities in either the teacher or student module to derive unified cell embeddings. **c**, Learned cell embeddings are fed into various downstream tasks (rigid annotation, clustering, data integration and query-to-reference mapping). See the Methods for detailed descriptions; all evaluated tasks are listed in Supplementary Table 9.

Concerto enables effective unsupervised clustering over multimodal dataset and can automatically extract molecular signatures from attention weights at single-cell resolution. A new single-cell study often starts with unsupervised clustering; however, discrete clusters might ignore smooth transitions among cell states. Cell-ID[44] can extract per-cell gene signatures in a clustering-free manner. Here we assess the utility of Concerto embeddings for de novo clustering and show that Concerto can also extract biologically meaningful

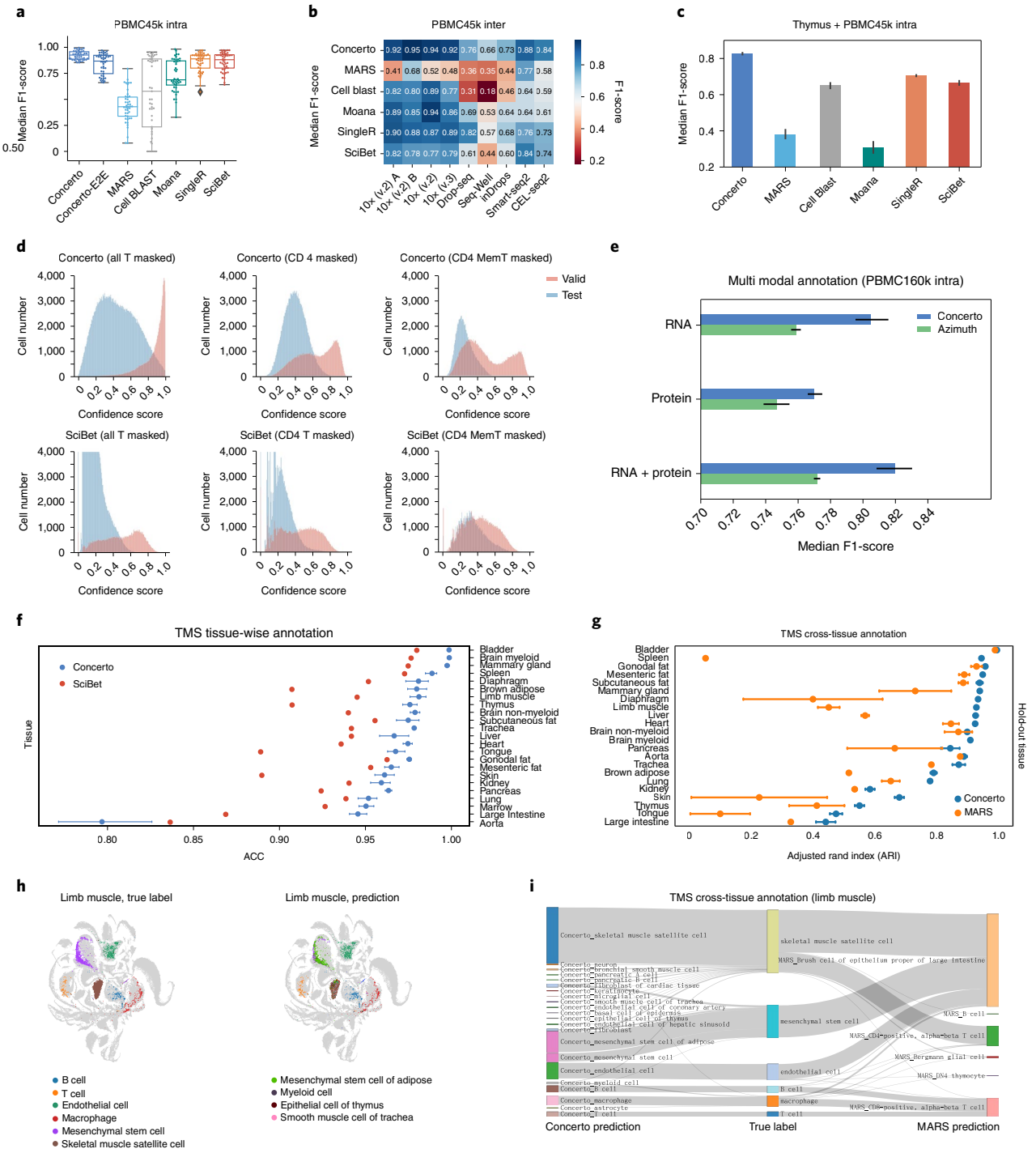


Fig. 2 | Contrastively learned embeddings notably boost the performance of automatic cell classification via fine-tuning and support novel cell type discovery across tissues. a,**b**, Comparing the performance of intra-dataset (fivefold cross-validation, nine batches) (**a**) and inter-dataset (bootstrapping five times) (**b**) predictions, as measured by the median F1-score of cell type labels on the PBMC45k scRNA-seq dataset ($n = 31{,}021$ cells) in comparison with MARS, Cell BLAST, Moana, SingleR and SciBet. The box plots show the median, and the first and third quartile values. The whiskers extend to points that lie within 1.5-times the interquartile range of the lower and upper quartile; observations that fall outside of this range are displayed independently. Concerto-E2E denotes end-to-end supervised training using the same model architecture without contrastive loss. **c**, Benchmark performance measured by the median F1-score on a challenging dataset comprising thymus scRNA-seq ($n = 107{,}969$ cells) and PBMC45k (fivefold cross-validation) data. Error bars represent the 95% confidence interval. **d**, A rejection option study comparing Concerto's ability with SciBet's to assign a low confidence score for non-existing cell types in the training set (fivefold cross-validation). **e**, Performance comparison for multimodal annotation on the PBMC160k dataset (RNA, protein, and RNA + protein) against Azimuth (fivefold cross-validation). Error bars represent the 95% confidence interval. **f**, Comparison of intra-tissue prediction accuracy on the TMS dataset ($n = 101{,}045$ cells). Error bars represent the 95% confidence interval. **g**, Benchmarking cross-tissue prediction measured by the ARI against MARS (bootstrapping three times). Error bars represent the 95% confidence interval. **h**, UMAP visualization of true labels versus Concerto predictions for hold-out limb muscle tissue ($n = 3{,}855$ cells). **i**, Sankey plot showing the label transfer of relevant cell types across tissues (using limb muscle as an example).

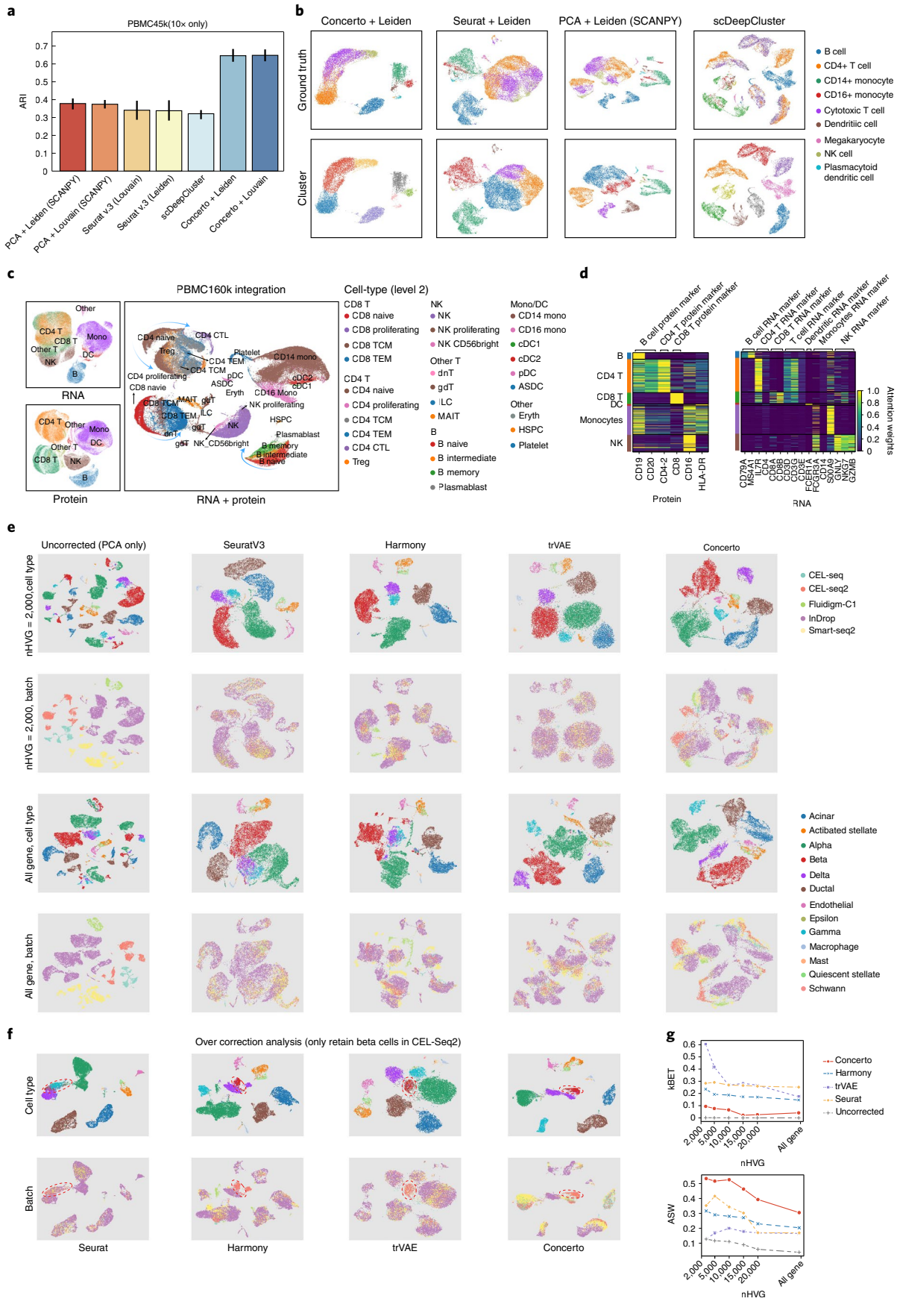
signatures at single-cell resolution. We choose a subset from PBMC45k (n = 11,377 cells, 10X protocols, 2,000 HVGs) with minimal batch effect. We compare Concerto's representations with Seurat's shared nearest neighbours on different clustering algorithms. scDeepCluster[45] is incorporated to represent simultaneous learning and clustering[46]. Three other deep learning methods are also evaluated: probabilistic-VAE scVI[11], graph neural network-based scGNN[47], and generative adversarial network-based scIGANs[48]. We use the normalized mutual information (NMI), ARI and silhouette score as evaluation metrics. Leiden clustering[49] on Concerto embeddings (Concerto + Leiden, mean NMI = 0.750, ARI = 0.646, silhouette score = 0.332) dramatically outperforms other methods across five resolutions (Fig. 3a, Extended Data Fig. 1a and Supplementary Table 12). Concerto well aligns cluster assignments with manual annotations (Fig. 3b; resolution = 0.4 for Leiden, k = 9 for scDeepCluster), clearly separating CD14 monocytes, CD16 monocytes and dendritic cells as different myeloid cells, and dividing the clear boundary between CD4 T cells and cytotoxic T cells. By contrast, other methods mix several populations (Supplementary Fig. 9–11 and 19). Benchmark results over another small dataset of mouse embryonic stem cells[50] (n = 2,000) can be found in Supplementary Table 13 and Concerto consistently performs the best.

To validate that incorporating other omics beyond the transcriptome[19] leads to a more precise definition of cell identity, we implement Concerto using RNAs, proteins, or both, as input and visualize learned embeddings coloured by hierarchical annotations (Fig. 3c). CD4 and CD8 T cells can be well separated by the proteins alone but partially mixed by RNAs alone. Natural killer (NK) cells are partially mixed with CD8 T cells by RNAs alone, but lie between other T cells and CD8 T cells for the protein-alone input. Dendritic cells are mingled with monocytes by proteins alone, but well separated by RNAs alone. These signals imply that proteins are more informative than RNAs for discriminating CD4 from CD8 T cells and uncovering subtle heterogeneity within NK cells. By contrast, the relationship of the monocytes and dendritic cell lineage can be better delineated by RNAs than proteins. Concerto displays a directional developmental trajectory (Fig. 3c) for the CD4 T cell lineage (CD4 naive, CD4 TCM and CD4 TEM), CD8 T cell lineage (CD8 naive, CD8 TCM and CD8 TEM) and B cell lineage (B naive, B intermediate, B memory and plasmablast). Furthermore, CD4+ regulatory T cells (Treg), MAIT cells and subpopulations of γδ-T cells (gdT) can be identified using dual-modality. Concerto can address any number of expanded modalities simply by implementing element-wise summation of each modality to obtain a unified view (Methods). We also show that Concerto obtains better clustering results (measured by ARI and NMI) than methods specifically designed for CITE-seq, including Seurat (WNN)[19], BREMSC[51], CiteFuse[52] and totalVI[53] (Extended Data Fig. 1 and Supplementary Table 11).

Concerto's teacher module uses the attention mechanism[30,31] to aggregate gene embeddings. We hypothesize that the attention weights might provide certain model interpretability by reproducing molecular signatures established for well-known cell types. Figure 3d and Supplementary Fig. 24 show the normalized attention contributions of key features to define cell identity, successfully recovering some canonical modality-specific markers for representative cell types (Methods). CD4 and CD8 T cells show divergent attention patterns of CD4 and CD8 protein markers, but no significant difference in their RNA transcripts, recapitulating the protein marker-based definition of these T cells. For B cells, CD19 protein and MS4A1 RNA (transcript of CD20 protein) emerge as key markers. Concerto also extracts the CD16 protein marker and cytotoxic RNA transcripts (GZMB, GNLY) in activated NK cells. Although neither clustering nor differential testing is used, some modality-specific signatures are automatically extracted by attention weights at single-cell resolution and match well with biological implications, representing a promising self-distillation marker identification protocol where the only learning signal is from each cell itself. We acknowledge this is a preliminary attempt and the robustness of attention weights should be further investigated. A similar idea was reported in DINO[29] to extract semantic layouts from natural images.

Concerto enables de novo data integration via removing unwanted batch effects and well supports integrating partially overlapping datasets. Facing the need to correct batch effects when combining different sources into a reference atlas, we benchmark Concerto's data integration performance on a well-curated multi-donor human pancreatic (HP) islet dataset (eight batches, five technologies, n = 14,890 cells)[54–58] against Seurat v.3, Harmony, trVAE and naive-PCA as baselines. Harmony and Seurat v.3 operate on principal components, whereas trVAE uses fully connected layers to compress the input. Concerto's encoding scheme can easily operate on all genes. We designed six scenarios to evaluate the impact of the number of input genes (Fig. 3 and Supplementary Fig. 12). All methods can combine different batches to varying degrees except for naive-PCA. We use the k-nearest-neighbour batch-effect test (kBET)[59] to quantify batch-mixing performance and the average silhouette width (ASW)[60] to evaluate cell type purity. Concerto achieves higher ASWs than competing methods by a large margin in the six scenarios (ASW = 0.533 for 2,000 HVGs, 0.305 for all genes; Fig. 3g), indicating better biological preservation. All methods show decreased ASW when accepting more genes as input, possibly because cell labels used to calculate ASW are derived from principal components (PCs) of 2,000 HVGs followed by manual inspection of cluster-specific signatures, more resembling the protocols used by Seurat and Harmony; 2,000 HVGs might not capture complete biological variations. Despite obtaining the lowest kBET score (0.10), Concerto successfully integrates eight sources at an acceptable level. We argue that no further mixing is necessary provided kBET reaches a certain threshold to ensure biological signals converge together rather than confounded by batch effects. Overpursuing a larger kBET might aggressively misalign distinct

Fig. 3 | Concerto enables effective unsupervised clustering over multimodal dataset and batch-corrected data integration. a, Evaluating Concerto embeddings on clustering performance against PCA, Seurat and scDeepCluster, as measured by the mean ARI over the PBMC45k dataset (10X only, n = 11,377 cells). Louvain and Leiden clustering methods are used for Concerto, PCA and Seurat (set five different resolutions at 0.1, 0.2, 0.3, 0.4, 0.6 for Louvain or Leiden; k = 7, 8, 9, 10, 11 for scDeepCluster). **b**, UMAP visualization of true cell type labels versus cluster assignment (resolution = 0.4 for Leiden; k = 9 for scDeepCluster). **c**, UMAP visualization of Concerto-learned embeddings on the PBMC160k dataset (RNA, protein, and RNA + protein), labelled by Azimuth (level-1 categories for unimodality and level-2 categories for dual-modality). The blue arrows show the directional distributions of subtypes within CD8 T cells, CD4 T cells and B cells. The black arrows are used to indicate some cell types, such as CD4 TCM, CD8 Naive, avoiding overlap and ambiguity in the figure. **d**, A heatmap showing how attention weights relate to some canonical modality-specific markers in major immune cell types. The ith row, jth column of the heatmap represents the attention weight of the ith cell's jth gene (or protein). **e**, UMAP visualization coloured by cell type label and batch label after integrating the human pancreatic islet scRNA-seq dataset (n = 14,890 cells, eight batches of five technologies). Benchmark methods include Seurat v.3, Harmony, trVAE and an uncorrected baseline (PCA only). **f**, Overcorrection analysis by removing all beta-cells (coloured red and indicated by red dashed ovals) except for CEL-Seq2, illustrated by UMAP visualization. **g**, Comparison of batch correction measured by kBET and ASW for six HVG scenarios (top 2,000, 5,000, 10,000, 15,000 and 20,000 HVGs, and all genes).


a
PBMC45k(10× only)
ARI
PCA + Leiden (SCANPY)
PCA + Louvain (SCANPY)
Seurat v.3 (Louvain)
Seurat v.3 (Leiden)
scDeepCluster
Concerto + Leiden
Concerto + Louvain
b
Concerto + Leiden
Seurat + Leiden
PCA + Leiden (SCANPY)
scDeepCluster
Ground truth
Cluster
B cell
CD4+ T cell
CD14+ monocyte
CD16+ monocyte
Cytotoxic T cell
Dendritiic cell
Megakaryocyte
NK cell
Plasmacytoid dendritic cell
c
PBMC160k integration
RNA
Protein
RNA + protein
Cell-type (level 2)
CD8 T
CD8 naive
CD8 proliferating
CD8 TCM
CD8 TEM
CD4 T
CD4 naive
CD4 proliferating
CD4 TCM
CD4 TEM
CD4 CTL
Treg
NK
NK
NK proliferating
NK CD56bright
Other T
dnT
gdT
ILC
MAIT
B
B naive
B intermediate
B memory
Plasmablast
Mono/DC
CD14 mono
CD16 mono
cDC1
cDC2
pDC
ASDC
Other
Eryth
HSPC
Platelet
d
B cell protein marker
CD4 T protein marker
CD8 T protein marker
B cell RNA marker
CD4 T RNA marker
CD8 T RNA marker
Dendritic RNA marker
Monocytes RNA marker
NK RNA marker
B
CD4 T
CD8 T
DC
Monocytes
NK
CD19
CD20
CD4-2
CD8
CD16
HLA-DR
Protein
RNA
Attention weights
e
Uncorrected (PCA only)
SeuratV3
Harmony
trVAE
Concerto
nHVG = 2,000,cell type
nHVG = 2,000, batch
All gene, cell type
All gene, batch
CEL-seq
CEL-seq2
Fluidigm-C1
InDrop
Smart-seq2
Acinar
Actibated stellate
Alpha
Beta
Delta
Ductal
Endothelial
Epsilon
Gamma
Macrophage
Mast
Quiescent stellate
Schwann
f
Over correction analysis (only retain beta cells in CEL-Seq2)
Cell type
Batch
Seurat
Harmony
trVAE
Concerto
g
kBET
nHVG
ASW
Concerto
Harmony
trVAE
Seurat
Uncorrected

cell populations, that is, overcorrection. To validate this hypothesis, we design a more complex scenario of integrating partially overlapping datasets by manually removing beta-cells from all five technologies except for CEL-Seq2. Harmony and Seurat mix up beta-cells with other types (Fig. 3f), implying overcorrection with lower beta-cell ASW but larger kBET (kBET = 0.32 and Beta-ASW = 0.29 for Harmony; kBET = 0.39 and Beta-ASW = 0.05 for Seurat v.3). Concerto clearly distinguishes beta-cells (Beta-ASW = 0.34 for Concerto), albeit obtaining a lower kBET value (0.03). To further justify Concerto's ability to avoid overcorrection, we design another controlled experiment using a simulated dataset (n = 12,097 cells, six batches of seven cell types, 2,000 HVGs) from a benchmark study (see the Supplementary Notes for details)[61]. We remove cell type-2 cells from all six batches except batch 1 to construct partially overlapping dataset before integration. Concerto clearly separates cell type-2 cells from other types (Extended Data Fig. 1c), obtaining larger cell type-2 ASW than other methods (kBET = 0.21 and cell type-2 ASW = 0.12 for Concerto; kBET = 0.43 and cell type-2 ASW = 0.06 for Harmony; kBET = 0.33 and cell type-2 ASW = 0.05 for Seurat V3; kBET = 0.21 and cell type-2-ASW = 0.09 for trVAE). Concerto's contrastive learning objective is immune to merging distinct subpopulations and preserves biological variation to build a high-quality reference.

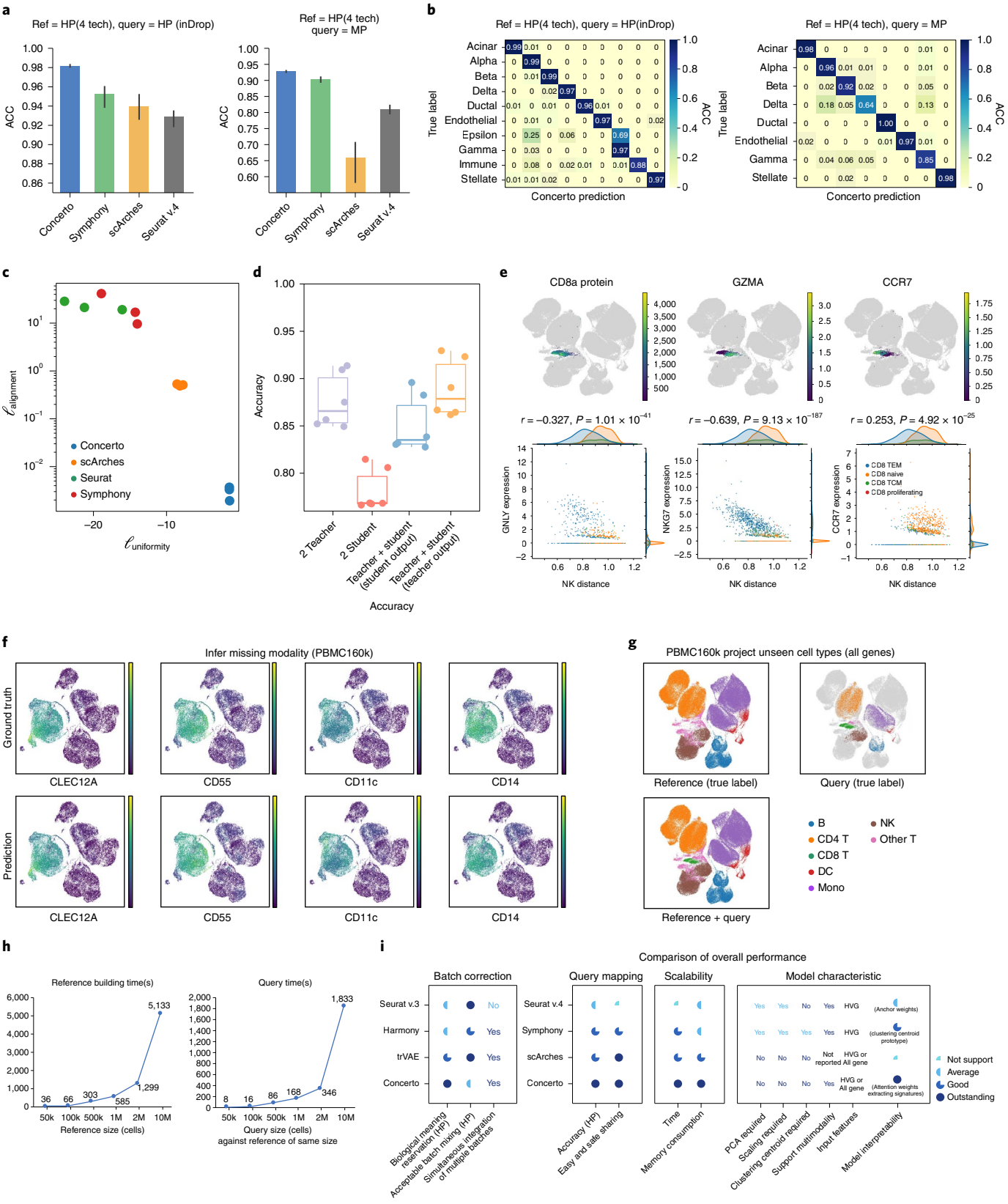
Concerto achieves state-of-the-art accuracy for query-to-reference mapping and supports projecting unseen cell types in the reference. We further evaluate Concerto for mapping query cells onto harmonized reference embeddings. Unlike rigid cell classification, query-to-reference mapping only uses cell type labels during inference. In particular, we first calculate query embeddings using pretrained model weights, locate query cells near their most similar reference cells and use a k-nearest-neighbour (usually k = 5) voting classifier to transfer reference annotations to queries. We design two experiments: cross-technology mapping, using the inDrop Baron dataset (n = 8,569 cells) as a query and all four other technologies (n = 6,321 cells) as a reference (HP → inDrop); and cross-species mapping using the same reference but with the mouse pancreatic islet (MP) (n = 1,880 cells) as a query (HP → MP). We benchmark against scArches, Symphony and Seurat v.4, with each corresponding to a reference building protocol (trVAE, Harmony and Seurat v.3, respectively); 2,000 HVGs are used for fair comparison. Concerto achieves the highest mean ACC for both experiments (0.981 for HP → inDrop, 0.927 for HP → MP, five replicates) (Fig. 4a). The confusion matrix (Fig. 4b) shows that Concerto can accurately transfer labels across technologies and species. Inspired by using uniformity and alignment to understand contrastive learning[23], we calculate these two properties (Methods) to explore the underpinnings of Concerto's superior performance in the query-to-reference mapping task. By comparison with the cell representations of other methods (Fig. 4c and Supplementary Notes), Concerto achieves the best alignment, which is two to four orders of magnitude lower than other methods (the lower the better), suggesting it has more consistent embedding quality to pull similar cells together on the hypersphere. We also conduct comprehensive ablation studies to justify why Concerto's asymmetric self-distillation architecture with a teacher network as output achieves better query-to-reference mapping performance than symmetric design (Fig. 4d and Supplementary Notes). Other components regarding hyperparameter choice, data-augmentation strategies, distillation schemes and the network structure for batch-effect correction are described in Supplementary Notes, with ablation results in Supplementary Figs. 13–17 and Supplementary Tables 4–6 and 8.

We then design a study to project unseen cell types and evaluate whether incorporating all genes can bring benefits. We assign one sample (P3) from PBMC160k as a query and use the other seven samples to build a reference. All CD8 T cells are removed from the reference. Concerto operating on all genes obtains considerably higher ACC (0.988 for all genes, 0.772 for 2,000 HVGs) and precisely localizes CD8 T cells between NK cells and CD4 T cells (Fig. 4g). Although Concerto has never seen CD8 T cells, the enrichment region of the CD8a protein marker overlaps with the positions of query CD8 T cells assigned by Concerto (Fig. 4e). Operating on all genes is expected to capture biological nuance to better identify fine-grained subtypes within CD8 T cells. The enrichment region of the cytotoxic marker GZMA is closer to NK cells, whereas the naive/memory-like marker CCR7 is located further away. Furthermore, transcriptional gradients of canonical markers correlate well with distances between CD8 T subtypes and NK cells. In the all-genes scenario, CD8 naive cells, which are further away from NK cells, show lower cytotoxic signatures (GNLY and NKG7) than proliferating and effector cells, whose locations are closer to NK cells (Fig. 4e), as quantified by negative Pearson correlation coefficients (r = −0.327 for GNLY, P-value = 1.01×10^{-41}; r = −0.639 for NKG7, P-value = 9.13×10^{-187}). The expression of naive signatures (CCR7) in CD8 T cells shows a positive correlation with the distance from NK cells (r = 0.253, P-value = 4.92×10^{-25}). We demonstrate that Concerto can project unseen cell subtypes along a biologically meaningful continuum. We also show that Concerto can infer unmeasured modalities in query cells. We leverage 80% of the samples from PBMC160k cells to build a dual-modality reference and use the remaining 20% with only the RNA count as a query (Methods). Concerto achieves consistent protein expression prediction against actual measurement in query cells (top-20 prediction, Pearson r = 0.966–0.998; Supplementary Figs. 22 and 23). Inferred

Fig. 4 | Concerto achieves state-of-the-art accuracy for query-to-reference mapping and supports projecting unseen cell types in the reference with multimillion-cell scalability. a, Performance comparison of query-to-reference mapping against Symphony, scArches and Seurat v.4. Left: HP → inDrop, n = 8,569 cells; the HP dataset is the reference except for when inDrop is used as the query. Right, HP → MP, n = 1,880 cells; the HP dataset is the reference except for inDrop and MP are used as queries (repeated five times). Error bars represent the 95% confidence interval. **b**, Confusion matrices of Concerto prediction measured by ACC. **c**, Alignment–uniformity plot for Concerto, scArches, Seurat and Symphony on HP dataset (HVGs = 2,000). There are three replicates (represented by dots) for each method. **d**, Ablation study on asymmetrical teacher–student network architectures; 2 teacher or 2 student represent both networks using attention operations or dense operations, respectively. Teacher + student denotes asymmetric design with the final cell embeddings extracted from the network and indicated in parentheses. The box plots show the accuracy of query-to-reference mapping for HP datasets (all genes, across six different k-values in nearest-neighbour-voting, k = 3, 5, 10, 15, 20, 25). **e**, Top: a heatmap showing that Concerto can successfully identify CD8 T cells masked in the reference, expressing the canonical CD8 protein marker, cytotoxic GZMA, and the CCR7 RNA marker enriched in the annotated CD8 T cell region. Bottom, Concerto preserves biological signals, showing a negative correlation of cytotoxic markers GMLY and NKG7, and a positive correlation of naive/memory marker CCR7 with the distance between CD8 T cells and NK cells (negative means increased expression at a closer distance to NK cells; positive means increased expression at a greater distance to NK cells). **f**, Heatmap of ground truth protein expression versus Concerto prediction for CLEC12A, CD55, CD11c and CD14 proteins visualized by UMAP (5-NN are used to infer L2-normalized protein expression). **g**, Illustration of Concerto's ability to project unseen cell types onto a reference by operating on all genes evaluated on the multimodal PBMC160k dataset (RNA + protein). All CD8 T cells in the query set are removed. **h**, Scalability of Concerto measured by elapsed time for reference building and querying cells of the same size. **i**, Schematic comparison among Concerto and three other mainstream packages.

expression paired with the ground truth of CLEC12A, CD55, and CD11c and CD14 are visualized by UMAP (Fig. 4f). Concerto shows great potential to uncover missing signals toward a holistic view of query cells.

Concerto can efficiently scale to 10-million-cell atlas construction and reference mapping. For scalability analysis, we simulate virtual references and map an equal number of query cells against each reference. Contrastive learning is naturally parallelizable and



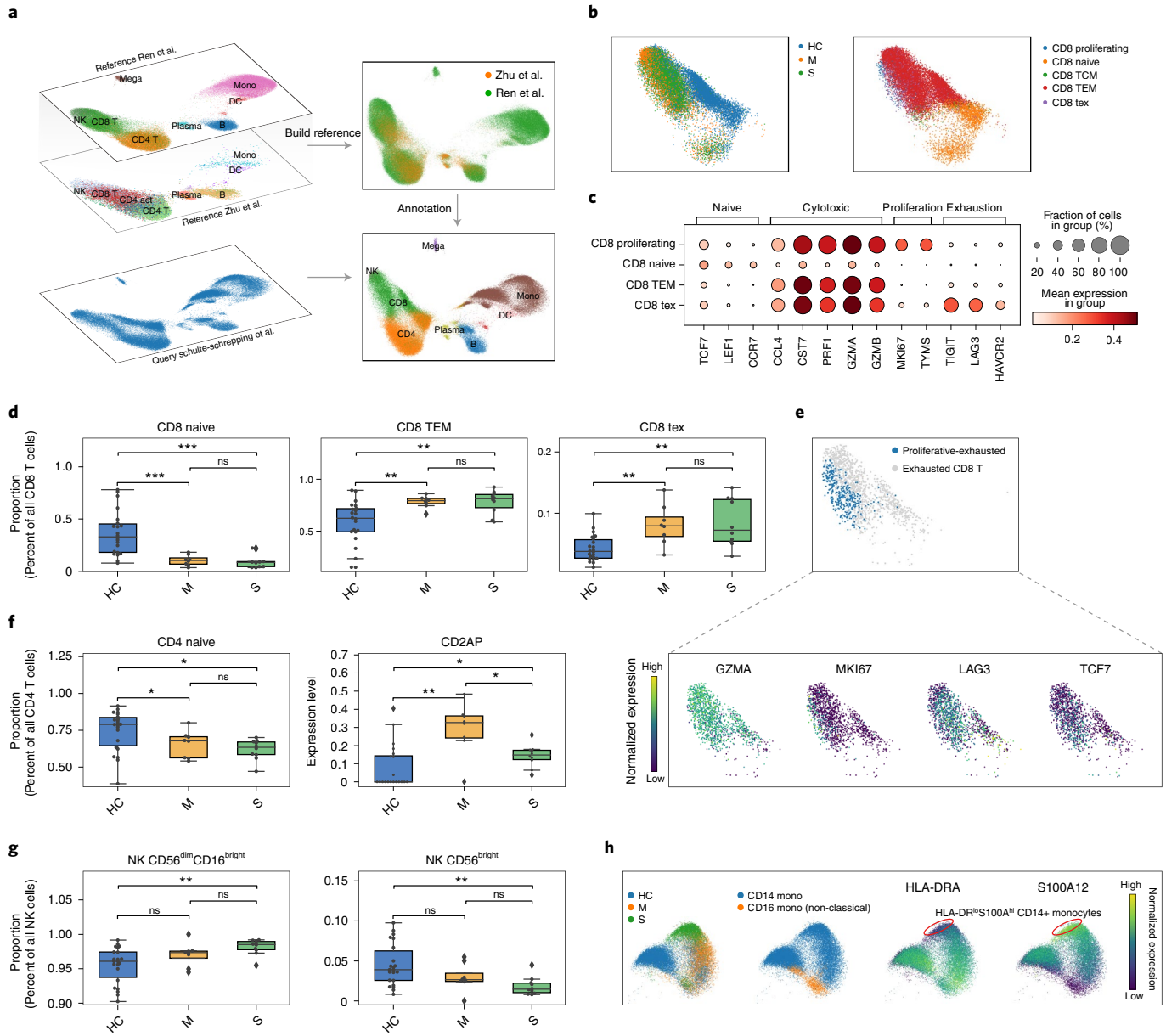


Fig. 5 | Hierarchical query-to-reference mapping preserves the differential immune response in COVID-19 patients. a, Illustration of mapping Schulte-Schrepping and colleagues'[64] PBMC scRNA dataset ($n = 99{,}049$ cells) onto an integrated COVID-19 reference (Ren et al.[62] and Zhu et al.[63]). **b**, UMAP visualization of annotated CD8 T cells divided into five subtypes and differential compositions among healthy controls (HC), moderate (M) and severe (S) disease statuses. **c**, Expression heatmap shows canonical markers for CD8 naive, cytotoxic, proliferating and exhaustion states. **d**, The box plots show the relative percentages of CD8 T cell subtypes among CD8 T cells at different disease statuses. **e**, UMAP shows proliferative-exhausted CD8 T cells and other exhausted CD8 T cells (top). Heatmap shows function-specific canonical markers in UMAP visualization (bottom). **f**, The box plots show the relative percentages of CD4 naive T cells among CD4 T cells at different disease statuses (left) and expression levels of T cell activation related genes in activated CD4 T cells at different disease statuses (right). **g**, The box plots show the relative percentages of NK CD56^{dim}CD16^{bright} and NK CD56^{bright} cells among annotated NK cells at different disease statuses. **h**, UMAPs show deficiency of an antigen presentation marker (HLA-DR) and enrichment of an inflammatory marker (S100A) that co-localize in the upper-left region of annotated monocytes, marked by red ovals. The box plots show the median, the first quartile and the third quartile values. The whiskers extend to points that lie within 1.5-times the interquartile ranges of the lower and upper quartiles and then observations that fall outside of this range are displayed independently. Mann–Whitney U test for differential expression analysis, two-sided, $^{*}P < 0.05$, $^{**}P < 0.01$, $^{***}P < 0.001$, $^{****}P < 0.0001$.

easily scalable to an extra-large atlas by dividing the whole task into multiple processing batches (Supplementary Table 10). By distributed training on eight orchestrated GPUs (NVIDIA Quadro RTX-6000), Concerto can build a 1-million-cell reference in 585 s (less than 10 min) and a 10-million-cell reference in 5,133 s (less than 1.5 h) (Fig. 4h). The reference only needs to be built locally and easily shared by model weights without compromising data privacy. Researchers can simply download pretrained reference weights and use in-house data to make direct inferences or perform unsupervised fine-tuning. Mapping a million-scale query takes 168 s (less than 3 min) (Fig. 4h). The peak memory usage is set to 6 GB per CPU (Intel Xeon Gold 6226 R) and 2.5 GB per GPU. Concerto can

efficiently scale to build a multimillion-cell reference, enabling rapid mapping within minutes. Concerto can also work on a typical computer using a CPU alone, taking 1.1 h to build a reference of 100,000 cells and query an equal number of cells within 30 min (Supplementary Fig. 21). We draw a schematic diagram to compare Concerto with other well-recognized tools (Fig. 4i). Concerto is the most scalable, does not require PCA or scaling, can operate on all genes and well supports multimodal integration.

Mapping COVID-19 immune cells against disease references reveals differential immune responses at different infection statuses. We further use Concerto to project a recently published COVID-19 PBMC dataset (Schulte-Schrepping et al., $n=99{,}049$) onto a comprehensive COVID-19 reference, which is built by integrating cells from Ren et al.[62] ($n=451{,}096$ PBMCs; 10X Chromium) and Zhu et al.[63] ($n=42{,}752$; DNBelab-C4 system). We then project query set (Schulte-Schrepping et al.[64]) onto it without fine-tuning (Fig. 5a). The COVID-19 reference contains disease-relevant cell states similar to those in the query; therefore, direct model inference is sufficient for rapid mapping. We propose a hierarchical mapping approach to enable effective interpretation. First, all query cells are mapped on top of the reference to obtain coarse-grained level-1 annotations, grouped query cells are then projected to subgroups of the reference to yield level-2 annotations. Schulte-Schrepping and colleagues' work focuses on myeloid cells, and we complement their analysis for lymphoid cells. Concerto can successfully localize query cells to obtain consistent level-1 subpopulations with the reference and identify perturbed pathological states through level-2 mapping (Fig. 5a).

For all annotated CD8 T cells, Concerto discriminates divergent compositions of naive, proliferating, memory and effector states at different disease states (healthy controls, mild and severe) and obtains concordant state-specific signatures (Fig. 5b,c and Methods). Naive markers are upregulated in annotated CD8 naive T cells (CCR7, LEF1, TCF7, SELL; logarithmic fold change ($\log_2$FC) = 0.98, 0.89, 0.81 and 0.30; false-discovery rate adjusted P-value (P_{adjusted}) = 2.2×10^{-308}, 3.0×10^{-263}, 8.6×10^{-92} and 5.8×10^{-11}, respectively; Wilcoxon rank-sum test), while the relative abundance of CD8 naive T cells significantly decreases in patients (Fig. 5d). CD8 T effector memory cells (TEM) manifest upregulated cytotoxic transcripts (PRF1, $\log_2\text{FC}=1.22$, $P_{\text{adjusted}}=2.2\times10^{-308}$; GNLY, $\log_2\text{FC}=1.28$, $P_{\text{adjusted}}=2.2\times10^{-308}$). Concerto also identifies emerging exhausted T cells in patient regions with increased exhaustion scores (such as LAG3; see Methods). We also validate the presence of a hybrid proliferative-exhausted CD8 T cell phenotype reported by Su et al.[65], co-expressing upregulated exhaustion transcripts (LAG3), proliferative marker (MK167) and cytotoxic signature (GZMA) without completely losing naive (TCF7) features (Fig. 5e).

For CD4 T cells, the relative abundance of CD4 naive T cells significantly decreases in COVID-19 patients (Fig. 5f), whereas the abundance of activated CD4 T cells increases in patients (Supplementary Fig. 20a)[65,66]. In particular, we annotate an activated CD4 T cell subtype with elevated CD2AP expression, indicating a dramatic state transition following infection, as presented by Zhu et al.[63] (Fig. 5f). CD2AP modulates the differentiation of follicular helper T cells, probably leading to an improved antibody response[67]. The proportion of regulatory T cells (Treg) increases in COVID-19 patients compared to healthy controls (Supplementary Fig. 20b), suggesting possible immunosuppression and an active anti-inflammatory response[68].

For NK cells, Concerto identifies $\text{CD56}^{\text{dim}}\text{CD16}^{\text{bright}}$ subpopulations that are significantly activated in severe patients (Fig. 5g; reported in another flow cytometry study[69]), showing elevated expression of cytotoxic markers (PRF1, $\log_2\text{FC}=0.70$, $P_{\text{adjusted}}=1.5\times10^{-16}$; GZMB, $\log_2\text{FC}=0.75$, $P_{\text{adjusted}}=6.5\times10^{-20}$) and exhaustion markers (HAVCR2, $\log_2\text{FC}=0.29$, $P_{\text{adjusted}}=2.1\times10^{-2}$). For monocytes, Concerto clearly separates healthy, moderate and severe samples (Fig. 5h). Non-classical monocytes ($\text{CD14}^{\text{low}}\text{CD16}^{\text{high}}$) are enriched in healthy samples but depleted in severe samples. For classical monocytes ($\text{CD14}^{\text{high}}\text{CD16}^{\text{low}}$), Concerto identifies a dysfunctional $\text{HLA-DR}^{\text{lo}}\text{S100A}^{\text{hi}}$ CD14+ subtype enriched in severe patients, recapitulating its inflammatory phenotype with antigen presentation deficiency (Schulte-Schrepping et al.[64] and Ren et al.[62]).

Overall, Concerto successfully separates pathological states, preserves nuanced status-specific variation, and identifies differential immune signatures. Whether implementing direct inference or unsupervised fine-tuning depends on reference diversity and relevance to the query. A more comprehensive reference usually benefits mapping performance (Supplementary Fig. 18 and Supplementary Notes). Concerto can be shaped as a continuous learning framework by iteratively updating references to cover more diverse samples.

Discussion

Assuming each cell is different, Concerto learns high-quality cell representations by discriminating each cell from others. Based on comparing different theoretical foundations with PCA or VAE-based methods, contrastively learned embeddings are well suited to preserve biological nuance as quantified by better cell alignment score in the latent space[23] (Fig. 4c). Concerto supports operating on all genes, which is particularly important to ensure feature overlap between query and reference in mapping-based tasks[18]. Inspired by recent progress in natural language processing[33], Concerto pioneers introducing model-level data augmentations in the omics field without disrupting molecular input. Concerto's asymmetric self-distillation scheme strikes a balance between learning semantically rich representations from the teacher network's attention operation and good generalizability from the student network's dense output (Fig. 4d and Supplementary Notes). By interpreting attention weights, we conceptually show that Concerto can automatically extract some canonical molecular signatures at single-cell resolution and identify relative contributions of each modality to define cell identity. Query-to-reference mapping has become a new paradigm in single-cell analysis. Concerto's contrastive setting is easily parallelizable and supports direct inference or unsupervised fine-tuning depending on reference diversity or relevance. Simply through element-wise summation, Concerto effectively supports multiomics integration. We plan to deploy Concerto in perturbation analysis to enable rapid identification of altered cell states upon stimulation. Concerto also shows great potential in translational research when large-scale disease atlases are available.

Methods

Concerto uses both simulated and real single-cell RNA-seq and CITE-seq datasets, and implements several benchmarking tasks, as listed in Supplementary Table 9. A full description of the data source can be found in Supplementary Table 1, with preprocessing details in Supplementary Table 2.

Overview of Concerto architecture. Concerto leverages an asymmetric self-distillation contrastive learning framework. The teacher module aggregates distributional gene embeddings using an attention mechanism[30] followed by nonlinear fully connected layers to obtain the teacher view for each cell, whereas the student module feeds discrete gene counts into dense layers. This asymmetric configuration injects imbalanced complexity presented as teacher and student. For model input, the normalized gene-count matrix is transformed into the index–value format, where index refers to gene identity defined by a dictionary comprising all genes of a certain species and value refers to corresponding counts in a cell. This encoding scheme supports sparse high-dimensional input and improves computational efficiency. The teacher module scales each gene's embedding by the corresponding count value. By defining a pretext task of discriminating each unlabelled cell, Concerto learns cell representations by maximizing agreement between each cell's different views using a contrastive loss in the latent space. Two augmented views for the same cell are obtained by passing the same cell through the student and teacher modules with a random dropout mask right before the output layer. Common perturbation techniques such as explicit transformations or randomness injections to original data might

alter the biological meaning. The dropout mask can be regarded as a minimal data augmentation without changing the original input[33]. Projected onto a unit hypersphere space[23], the contrastive loss compares pairs of cell embeddings by pushing apart different cells within a batch while pulling together teacher–student views of the same cell as positive pairs. The distance is measured by the cosine similarity of L2-normalized embeddings using the dot product operation. To process multiomics data, simple element-wise summation of modal-specific attention output in the teacher module or dense output in the student module enables the generation of a unified cell view (Fig. 1b). Learned embeddings can then be fine-tuned for various downstream tasks, including automatic cell type classification, clustering, data integration for batch-effect correction, and query-to-reference mapping.

1. For automatic cell type identification, Concerto implements task-agnostic pretraining followed by supervised fine-tuning using existing annotations. For the within-datasets (intra-dataset) prediction, we fine-tuned Concerto via an extra fully connected classification layer with a softmax operation over the dimensions of the predefined categories. For cross-datasets (inter-dataset) prediction, we conduct semi-supervised fine-tuning by adding a domain adaptation module.
2. To group functionally similar cells into clusters, Concerto decouples cell representation learning and clustering into two stages, which are expected to be less sensitive to model initialization than one-step approaches.
3. For de novo data integration, Concerto aims to learn batch-invariant embeddings across species, technologies, experimental conditions or sample status. These metadata are incorporated as model input to guide source-specific batch normalization[70] within a training mini-batch. This simple configuration enables Concerto to extract a batch-invariance biological signal[34] to remove unwanted confounding factors.
4. A reference atlas is constructed for query-to-reference mapping. Query cell embeddings are simply inferred by passing them through the trained teacher network. In this case, users directly utilize reference model weights and contextualize query cells onto a stable reference space. Reference annotations can be easily transferred to query cells through an nearest-neighbour voting scheme to derive a fast interpretation. This task is distinct from supervised rigid annotation as in part 1, that is, the cell type labels are never used in the training process. On the other hand, users can also leverage reference weights as model initialization and implement unsupervised fine-tuning on query cells. Concerto can be continuously updated to construct a more comprehensive atlas.

Filtering, preprocessing and normalization. For scRNA-seq dataset, we delete mitochondrial genes (ERCC, MT-, mt-), discard low-quality cells with fewer than 600 genes and remove genes expressed in fewer than three cells. We use SCANPY (v.1.7.1) to normalize each cell count to 10,000 read counts before logarithmic transformation. For protein, original data are used except for missing modality inference.

HVG selection. Concerto supports operating on both all genes and selected HVGs by SCANPY[4] (1.7.1). For the all-genes scenario, the same number of genes is used within a batch, whereas for the HVG scenario, only selected HVGs are used to generate the index and value.

Homologue alignment. In the HP → MP transfer task, orthologous genes in the Mouse Genome Informatics database are used[71].

Input encoding scheme. TensorFlow Record (TF-record) file is used to encode the normalized gene-count matrix. TF-record is a binary file containing sequences of serialized byte strings for the sharding file in TensorFlow. Concerto encapsulates 'gene index' and 'count value' into the TF-record file. The teacher network accepts both the gene index and count value, whereas the student network reads only the count value file.

Teacher network. The teacher network accepts $X_{\text{indices}} \in \mathbb{R}^G$ and $X_{\text{counts}} \in \mathbb{R}^G$, where G denotes the number of genes. X_{indices} represents gene indices, where indices refer to the gene identity defined by a dictionary comprising all genes of a certain species. Each gene within a cell is represented by i, where $i \in \mathbb{R}^G$. X_{counts} represents the value of gene counts. Embedding denotes a neural network with only one fully connected layer to transform a sequence of a cell's molecular input to learned embedding. First, $\mathbf{x}_{\mathbf{indices}}$ is embedded into a d-dimensional vector space $\mathbf{emb}$, $\mathbf{emb}_i \in \mathbb{R}^d$(1), where d is set to 128 as the default. X_{indices} is a matrix with N*G dimension while $\mathbf{x}_{\mathbf{indices}}$ is a vector. X_{indices} is the collection of all $\mathbf{x}_{\mathbf{indices}}$. The cross product of emb and $\mathbf{x}_{\mathbf{counts}}$ outputs the weighted hidden vector, $\mathbf{hidden}_i, \mathbf{hidden}_i \in \mathbb{R}^d$, see equation (2)

$$\mathbf{emb} = \text{Embedding}\,(\mathbf{x}_{\mathbf{indices}}) \tag{1}$$

$$\mathbf{hidden}_i = \mathbf{emb} \times \mathbf{x}_{\mathbf{counts}} \tag{2}$$

Concerto then uses attention mechanism[31] to aggregate gene embeddings; $\mathbf{hidden}_i$ is first fed into an multilayer perceptron with one hidden layer, and a nonlinear tanh transformation is activated to obtain a hidden vector, $\mathbf{hidden}_i$ (equation (3)). A cellular context vector $\mathbf{u} \in \mathbb{R}^d$ then applies the dot product to $\mathbf{hidden}_i$, using the softmax operation to obtain attention weights, $\mathbf{attention}_i \in \mathbb{R}^G$ (equation (4)); aggregation is then implemented on all of the genes' vectors $\mathbf{hidden}_i$ through weighted summation by attention weights, $\mathbf{attention}_i$, to obtain aggregated vectors, $\mathbf{hidden}$ (equation (5)).

$$\mathbf{hidden}_i = \tanh(\mathbf{hidden}_i) \tag{3}$$

$$\mathbf{attention}_i = \text{softmax}(\mathbf{hidden}_i \cdot \mathbf{u}) \tag{4}$$

$$\mathbf{hidden} = \sum_i (\mathbf{attention}_i \times \mathbf{hidden}_i) \tag{5}$$

The attention layer output is fed into a batch normalization layer followed by a dropout layer, and then a dense layer with ReLU activation leads to the final output of the teacher network, $Z_{\text{teacher}} \in \mathbb{R}^d$ (equation (6))

$$\mathbf{z}_{\mathbf{teacher}} = \text{BatchNormalization}(\mathbf{hidden}) \tag{6}$$

$$\mathbf{z}_{\mathbf{teacher}} = \text{Dropout}(\mathbf{z}_{\mathbf{teacher}}) \tag{7}$$

$$\mathbf{z}_{\mathbf{teacher}} = \text{ReLU}(\mathbf{z}_{\mathbf{teacher}}) \tag{8}$$

Student network. The student network accepts only $X_{\text{counts}} \in \mathbb{R}^G$, then going through a batch normalization layer followed by a dropout layer and then a dense layer with ReLU activation, leading to the final output of the student network, $Z_{\text{student}} \in \mathbb{R}^d$.

$$\mathbf{hidden} = \text{Dense}(\mathbf{x}_{\mathbf{counts}}) \tag{9}$$

$$\mathbf{hidden} = \text{BatchNormalization}(\mathbf{hidden}) \tag{10}$$

$$\mathbf{hidden} = \text{Dropout}(\mathbf{hidden}) \tag{11}$$

$$\mathbf{z}_{\mathbf{student}} = \text{ReLU}(\mathbf{hidden}) \tag{12}$$

Data augmentation with a dropout layer. The dropout layer operation is used as a model-level data augmentation strategy[33]. By randomly masking neural units with a certain probability (parameters, dropout rate = 0.2) before the final dense layer, augmented embeddings of the same cell are generated for contrastive learning.

Contrastive loss (NT-Xent loss). Contrastive learning is conducted on a unit hypersphere space and explicitly compares pairs of cell embeddings of d dimension (where $d = 128$ by default). Through contrastive learning, for each cell, Concerto pushes apart all other cells and their augmentations within a batch while pulling together teacher–student views of the same cell as positive pairs. As we use an asymmetric teacher–student network, we obtain two different embeddings, $\mathbf{z}_{\mathbf{teacher}}$ and $\mathbf{z}_{\mathbf{student}}$.

The distance of the two given embeddings is defined by equations (13) and (14). Assume a positive pair as cell_j (which embeds $\mathbf{z}_{\mathbf{teacher}_j} \in Z_{\text{teacher}}$) and cell_{j+} (which embeds $\mathbf{z}_{\mathbf{student}_{j+}} \in Z_{\text{student}}$), where τ is the adjustable temperature coefficient, which can be used to scale the degree of pushing apart negative samples. NT-Xent loss represents the normalized temperature-scaled cross-entropy loss, as formalized by equation (15), where j and j^+ is a pair of positive samples. We randomly sample a mini-batch of N cells and compute NT-Xent loss on pairs of augmented examples derived from the mini-batch, resulting in $2N$ data points. Given a positive pair, the other $2(N-1)$-augmented examples within a mini-batch are treated as negative examples.

$$s_{\alpha,\beta} = \mathbf{z}^T_{\mathbf{teacher}_\alpha} \mathbf{z}_{\mathbf{student}_\beta} / \tau \parallel \mathbf{z}_{\mathbf{teacher}_\alpha} \parallel \parallel \mathbf{z}_{\mathbf{student}_\beta} \parallel \tag{13}$$

$$s^+_{\alpha,\beta} = \mathbf{z}^T_{\mathbf{student}_\alpha} \mathbf{z}_{\mathbf{teacher}_\beta} / \tau \parallel \mathbf{z}_{\mathbf{student}_\alpha} \parallel\parallel \mathbf{z}_{\mathbf{teacher}_\beta} \parallel \tag{14}$$

$$\mathcal{L} = \frac{1}{2N} \sum_{k=1}^{N} [\ell\left(j, j^+\right) + \ell(j^+, j)] \tag{15}$$

where $\ell\left(j, j^+\right)$ is defined as:

$$\ell\left(j, j^+\right) = -\log \frac{\exp\left(s_{j,j+}\right)}{\sum_{k=1}^{2N} \mathbb{I}_{[k \neq j]} [\exp\left(s_{k,j}\right) + \exp\left(s_{k,j+}\right)]}$$

where $\ell\left(j^+, j\right)$ is defined as:

$$\ell\left(j^+, j\right) = -\log \frac{\exp\left(s^+_{j^+,j}\right)}{\sum_{k=1}^{2N} \mathbb{I}_{[k\neq j^+]}\left[\exp\left(s^+_{k,j^+}\right) + \exp\left(s^+_{k,j}\right)\right]}$$

Multimodal integration. Concerto supports convenient multimodal integration. To process multiomics datasets, simple element-wise summation of modal-specific attention output in the teacher module or dense output in the student module enables Concerto to generate unified cell embeddings. In the case of two modalities (RNA and protein), we illustrate the respective operations as per equations (16) and (17), where the term Add denotes add along the dimension of embedding.

$$\mathbf{z}^{\text{multi}}_{\text{student}} = \text{Add}\left(\mathbf{z}^{\text{RNA}}_{\text{student}}, \mathbf{z}^{\text{protein}}_{\text{student}}\right) \quad \mathbf{z}^{\text{multi}}_{\text{student}} \in \mathbb{R}^d \tag{16}$$

$$\mathbf{z}^{\text{multi}}_{\text{teacher}} = \text{Add}(\mathbf{z}^{\text{RNA}}_{\text{teacher}}, \mathbf{z}^{\text{protein}}_{\text{teacher}}) \quad \mathbf{z}^{\text{multi}}_{\text{teacher}} \in \mathbb{R}^d \tag{17}$$

Pretraining procedure. We input cells $\mathbf{x} \in X_{\text{indices}}, X_{\text{counts}}$ into the asymmetric teacher–student network to obtain Z_{teacher} and Z_{student} and then project them onto the unit hypersphere space. We use the contractive loss to explicitly compare pairs of cell embeddings and maximize the agreement between teacher–student views of the same cell so that the contrastively learned embeddings are expected to capture high-level features to discriminate different cells for downstream usage. We denote $f_{\text{pre-training}}$ as the pretraining procedure, which is also shown in Fig. 1a.

Supervised fine-tuning. For rigid annotation, Concerto leverages contrastive learning as a task-agnostic pretraining procedure followed by supervised fine-tuning using manually annotated labels. For within-datasets prediction (intra-dataset), we fine-tune Concerto via an extra fully connected classification layer with a softmax operation over the dimensions of the predefined cell type categories. The loss function is the classical supervised cross-entropy loss (equation (18)). For the cross-dataset (inter-dataset) prediction, we conduct semi-supervised fine-tuning by adding a domain adaptation module to derive cross-tissue or cross-species predictions. We also validate the inferior performance of end-to-end training (Concerto-E2E) by discarding the contrastive loss without a self-supervised training procedure while retaining only the model backbone to conduct fully supervised training.

$$\min \mathcal{J}(\theta) = \mathbb{E}_{\mathbf{x}\sim PL(\mathbf{x})}\left[\text{CE}\left(p_\theta\left(f_{\text{pre-training}}(\mathbf{x})\right) \parallel x^*\right)\right] \tag{18}$$

where $\mathbf{x} \in X_{\text{indices}}, X_{\text{counts}}$ represents the input of Concerto and CE represents the cross-entropy method. We first conduct pretraining on $\mathbf{x}$ and then fine-tune Concerto via a fully connected layer with a softmax operation and output $p_\theta(\mathbf{x})$; $p_\theta(\mathbf{x})$ denotes the predicted classification probability; θ represents the parameters of the final classification layer; x^* represents the true labels of the input data; and PL $(\mathbf{x})$ represents the distribution of the input data. Furthermore, the confidence score of $\mathbf{x}$ is calculated by the min–max-scaled $p_\theta\left(f_{\text{pre-training}}(\mathbf{x})\right)$.

Domain adaptation module. For inter-dataset annotation, we add a domain adaptation module[43] to adapt the target labels to the source distribution. In addition to the supervised cross-entropy loss, we add an unsupervised consistency training loss (equation (19)).

$$\text{Min}\mathcal{J}(\theta) = \lambda \mathbb{E}_{\mathbf{y}\sim \text{PU}(\mathbf{y})} \mathbb{E}_{\tilde{\mathbf{y}}\sim \text{PU}(\tilde{\mathbf{y}})}[\text{CE}(p_{\tilde{\theta}}(\mathbf{y}) \parallel p_\theta(\tilde{\mathbf{y}}))] \tag{19}$$

where $\mathbf{y} \in X_{\text{indices}}, X_{\text{counts}}$ denotes unlabelled target data; $p_\theta(\mathbf{y})$ denotes the predicted classification probability; $\tilde{\theta}$ is a fixed copy of the current parameter θ, indicating that the gradient is not propagated through $\tilde{\theta}$; $\tilde{\mathbf{y}}$ represents augmented (via the dropout layer) unlabelled target data; and PU $(\mathbf{y})$ denotes target data distribution. We set λ to 1 to balance loss term from equations (18) and (19) to train Concerto in a semi-supervised setting for inter-dataset prediction. This module combines consistency loss with cross-entropy loss to align two data distributions from the source domain and target domain, which enables dealing with potential batch effects encountered in inter-dataset rigid annotation tasks.

Source-aware batch normalization for data integration. For data integration, Concerto aims to learn batch-invariance embeddings to integrate heterogeneous data sources and overcome batch effects[34]. Metadata of source information are used as source identities. Batch normalization is only conducted for cells from the same source within a training mini-batch.

Nearest-neighbour voting classifier. To transfer annotations from reference cells to query cells after obtaining all cell embeddings, Concerto uses a simple k-nearest-neighbour voting classifier to annotate query cells. The nearest-neigbour voting classifier assigns the k-nearest-neighbours for query cells $\mathbf{y}$, where $\mathbf{y} \in X_{\text{indices}}, X_{\text{counts}}$. For query cells $\mathbf{y}$, we extracted their k-nearest-neighbours (N_y). In D_{y,N_y} (equation (20)), cosine similarity is used to calculate the distance between $\mathbf{y}$ and their neighbours n in the latent space (cell embeddings). The normalization of D_{y,N_y} is implemented as in equation (21) to calculate $p(x^*, \mathbf{y})$, which is the probability of assigning reference annotations (x^*) to $\mathbf{y}$, where $x^{*(i)}$ is the annotation label of the ith neighbour and y' is the transferred annotation with the maximum probability (equation (22)). We set k to 5 for most cases, while k can be tuned accordingly. Furthermore, the confidence score of $\mathbf{y}$ is calculated by $p(x^*, \mathbf{y})$.

$$D_{y,N_y} = \text{cosine}(\mathbf{y}, n) \tag{20}$$

$$p\left(x^*, \mathbf{y}\right) = \frac{\sum_{i\in N_y} I(x^{*(i)} = x^*) D_{y,N_y}}{\sum_{i\in N_y} D_{y,N_y}} \tag{21}$$

$$y' = \text{argmax}(p\left(x^*, \mathbf{y}\right)) \tag{22}$$

UMAP visualization. Cell embeddings are visualized by UMAP using scanpy.pl.umap from SCANPY (v.1.7.1). The number of neighbours (n_neighbours) is set to 15; use_rep = X; and metric = euclidean. The other functions use the default parameters.

Hyperparameters. The learning rate in contrastive pretraining is set to varied values from 1×10^{-4} to 1×10^{-6} using Adam optimizer training for three epochs. For fine-tuning, the learning rate is set to 1×10^{-3} using Adam optimizer training for one epoch. The temperature coefficient in NT-Xent loss is set to 0.1, the mini-batch size is set to 32, and $d = 128$. Ablation studies are detailed in Supplementary Notes.

Attention weight extraction. Attention weights are calculated by the following steps: first, the cell's gene embeddings $\mathbf{hidden}_{i=1,2,\ldots,G}$ $(1\times G\times d)$ are fed into an multilayer perceptron with one hidden layer with nonlinear tanh transformation to obtain hidden vectors $\mathbf{hidden}_{i=1,2,\ldots,G}$ $(1\times G\times d)$; then, a cellular context vector $\mathbf{u}$ $(d\times 1)$ applies a dot product to $\mathbf{hidden}_{i=1,2,\ldots,G}$ using a softmax operation to obtain attention weights $\mathbf{attention}_i$ $(1\times G)$. All calculations are conducted in a 128-dimensional space.

Calculation of alignment and uniformity. The alignment (equation (23)) is simply defined as the expected distance between positive pairs $(\mathbf{x}, \mathbf{x}^+)$. First we conduct pretraining on input $\mathbf{x}$ and obtain the embedding $\mathbf{z}^{(\mathbf{x})}_{\mathbf{teacher}}$. To find $\mathbf{x}^+$, which is the positive pair of $\mathbf{x}$, we use k-nearest neighbours to extract the nearest neighbours of $\mathbf{x}$ (N_x), and we assign $k = 5$ nearest neighbours as $\mathbf{x}^+$.

$$\ell_{\text{align}} \triangleq \mathbb{E}_{(\mathbf{x},\mathbf{x}^+)\sim p_{\text{pos}}} \left\| \mathbf{z}^{(-)}_{\mathbf{teacher}} - \mathbf{z}^{(\mathbf{x}^+)}_{\mathbf{teacher}} \right\|^2 \tag{23}$$

where p_{pos} is a distribution of positive pairs.

The uniformity (equation (24)) is defined as the logarithm of the average pairwise Gaussian potential.

$$\ell_{\text{uniform}} \triangleq \log \mathbb{E}_{(\mathbf{x}_\alpha,\mathbf{x}_\beta)\sim p_{\text{data}}} \text{e}^{-2\left\| \mathbf{z}^{(\mathbf{x}_\alpha -)}_{\mathbf{teacher}} - \mathbf{z}^{(\mathbf{x}_\beta)}_{\mathbf{teacher}} \right\|^2} \quad (\alpha \neq \beta) \tag{24}$$

where $\mathbf{x}_\alpha$ and $\mathbf{x}_\beta$ denote different data, and p_{data} denotes the data distribution.

Missing modality inference. We first split the PBMC160k dataset into eight donors (where each donor represents one replicate). For each donor, we randomly sample 80% of the cells to build a multimodal reference and obtain 128-dimensional embedding for each reference cell. We then calculate query embeddings using only the RNA modality as input for the remaining 20% hold-out cells of each donor. It is noted that no fine-tuning process is needed. Based on the inferred query embeddings, we compute the expression of 224 surface proteins for query cells by averaging normalized protein expression across the cell's five nearest neighbours in the reference, whose distance is defined as cosine similarity in 128-dimensional embedding space.

NOTA evaluation. We evaluate whether Concerto can support NOTA cells as a rejection option if the test set contains certain cell subpopulations not existing in training samples. These cells cannot be accurately predicted and should be assigned as NOTA when the classifier is not confident enough to annotate them with predefined labels. We download a multimodal PBMC CITE-seq atlas of 161,764 cells (PBMC160k) with three levels of annotations. Only RNA counts are used as input features in this rejection study. For different levels, we remove different granularities of T cells from the training set to form progressively increasing difficulties. First, all T cells are removed; then only CD4 T cells are removed; and then only CD4 Mem T cells are removed. Twenty percent of the training set is randomly selected as the validation set. The test set only contains removed cell types at each level (detailed mask setting in Supplementary Table 3).

A qualified classifier should predict accurate labels for cells in the validation set while assigning NOTA to cells from the test set.

Robustness analysis. We use the R package Splatter[41] to simulate scRNA-seq data to mimic various biological scenarios under different dropout rates (defined as the proportion of expressed genes being knocked out) and different expression signal strengths (defined as various fold change levels of differential genes). For differential expression (DE) simulation, the following parameters are used in the splatSimulate() R function: groupCells = 5, nGenes = 2,500, dropout.present = TRUE, dropout.shape = 1, dropout.mid = 1, and de.scale = 0.15, 0.2, 0.25, 0.3, respectively. For dropout rate simulation, the following parameters are used in the splatSimulate() R function: groupCells = 5, nGenes = 2,500, dropout.present = TRUE, dropout.shape = 1, dropout.mid = −0.5, 0, 0.5, 1, respectively, and de.scale = 0.2.

Scalability analysis. For scalability analysis, simulated datasets are generated using the scsim Python package[72], which is based on the Splatter statistical framework, while it performs more efficiently to generate large-scale simulated data. The following parameters are used in the scsim() Python function: ngenes = 25,000; ncells = 50,000, 100,000, 500,000, 1,000,000, 2,000,000 and 10,000,000; ngroups = 5; diffexpprob = 0.025.

Exhausted T annotation in COVID-19 analysis. As exhausted CD8 T cells do not exist in the COVID-19 reference, we calculate the exhaustion signature score using an exhaustion gene set (PDCD1, TIGIT, LAG3, HAVCR2, CTLA4) by summing their expression values (scaled to 0 to 1). CD8 T cells with exhaustion scores of greater than 0.7 are annotated as exhausted T cells.

Identification of COVID-19 reference subpopulations. Ren et al.[62] and Zhu et al.[63] previously characterized cell type specific subpopulations and detailed the function of each subgroup. On the basis of collecting markers from their works, subtypes of CD8 T cells, CD4 T cells, NK cells and monocytes are identified following the functional description in their original papers.

Analytic metrics. *F1-score and ACC.* The F1-score and ACC are metrics for classification performance calculated by the Python functions sklearn.metrics.f1_score() and sklearn.metrics.accuracy_score() from the scikit-learn library, respectively.

ARI and NMI. The ARI and NMI are applied to assess clustering performance calculated by the Python functions adjusted_rand_score() and normalized_mutual_info_score() from the scikit-learn library, respectively.

ASW. The ASW is calculated using the Python function sklearn.metrics.cluster.silhouette_samples() from the scikit-learn library as an evaluation metric for biological meaning reservation.

kBET. kBET[61] is a metric for batch-effect correction, indicating how well mixed batches from randomly sampled nearest-neighbour cells are based on local batch label distribution consistent with global batch label distribution. Pegasus is adopted to calculate kBET, and k is set to 15.

Clustering. Python function sklearn.cluster. KMeans() is used to perform k-means clustering. To perform the Leiden and Louvain algorithm, we apply the R function FindClusters() from the R package Seurat v.3, and the parameter 'algorithm' is set to 4 and 1. We also apply the Python functions scanpy.tl.leiden and scanpytl.louvain from SCANPY in SCANPY-relevant clustering analysis.

Other benchmarking tools. Descriptions of implementing other benchmarking tools can be found in the Supplementary Notes.

Reporting summary. Further information on research design is available in the Nature Research Reporting Summary linked to this article.

Data availability

All of the scRNA-seq and CITE-seq datasets in this study were published previously[2,19,35,42,54–58,61–64]; their availabilities, alongside downloadable links, are described in Supplementary Table 1. We have uploaded four pre-built references with corresponding model weights learned by Concerto to facilitate community usage (see Supplementary Table 7). Source Data are provided with this paper[73].

Code availability

Concerto is written in Python using the TensorFlow library. The source code with reproducibility demo is available on Github at https://github.com/melobio/Concerto-reproducibility under the GPLv3 license (https://doi.org/10.6084/m9.figshare.19351745).

Received: 10 September 2021; Accepted: 11 July 2022;
Published online: 25 August 2022

References

1. Rozenblatt-Rosen, O., Stubbington, M. J. T., Regev, A. & Teichmann, S. A. The human cell atlas: from vision to reality. *Nature* **550**, 451–453 (2017).
2. Tabula Muris Consortium. A single cell transcriptomic atlas characterizes aging tissues in the mouse. *Nature* **583**, 590 (2019).
3. Satija, R., Farrell, J. A., Gennert, D., Schier, A. F. & Regev, A. Spatial reconstruction of single-cell gene expression data. *Nat. Biotechnol.* **33**, 495–502 (2015).
4. Wolf, F. A., Angerer, P. & Theis, F. J. SCANPY: large-scale single-cell gene expression data analysis. *Genome Biol.* **19**, 15–15 (2018).
5. Li, B. et al. Cumulus provides cloud-based data analysis for large-scale single-cell and single-nucleus RNA-seq. *Nat. Methods* **17**, 793–798 (2020).
6. Kiselev, V. Y., Andrews, T. S. & Hemberg, M. Challenges in unsupervised clustering of single-cell RNA-seq data. *Nat. Rev. Genet.* **20**, 273–282 (2019).
7. Tran, H. T. N. et al. A benchmark of batch-effect correction methods for single-cell RNA sequencing data. *Genome Biol.* **21**, 1–32 (2020).
8. Abdelaal, T. et al. A comparison of automatic cell identification methods for single-cell RNA sequencing data. *Genome Biol.* **20**, 1–19 (2019).
9. Svensson, V. Droplet scRNA-seq is not zero-inflated. *Nat. Biotechnol.* **38**, 147–150 (2020).
10. Abdi, H. & Williams, L. J. Principal component analysis. *Wiley Interdiscip. Rev. Comput. Stat.* **2**, 433–459 (2010).
11. Lopez, R., Regier, J., Cole, M. B., Jordan, M. I. & Yosef, N. Deep generative modeling for single-cell transcriptomics. *Nat. Methods* **15**, 1053–1058 (2018).
12. Zhao, Y., Cai, H., Zhang, Z., Tang, J. & Li, Y. Learning interpretable cellular and gene signature embeddings from single-cell transcriptomic data. *Nat. Commun.* **12**, 1–15 (2021).
13. Stuart, T. et al. Comprehensive integration of single-cell data. *Cell* **177**, 1888–1902 (2019).
14. Haghverdi, L., Lun, A. T. L., Morgan, M. D. & Marioni, J. C. Batch effects in single-cell RNA-sequencing data are corrected by matching mutual nearest neighbors. *Nat. Biotechnol.* **36**, 421–427 (2018).
15. Butler, A., Hoffman, P., Smibert, P., Papalexi, E. & Satija, R. Integrating single-cell transcriptomic data across different conditions, technologies, and species. *Nat. Biotechnol.* **36**, 411–420 (2018).
16. Korsunsky, I. et al. Fast, sensitive and accurate integration of single-cell data with Harmony. *Nat. Methods* **16**, 1289–1296 (2019).
17. Lotfollahi, M., Naghipourfar, M., Theis, F. J. & Wolf, F. A. Conditional out-of-distribution generation for unpaired data using transfer VAE. *Bioinformatics* **36**, i610–i617 (2020).
18. Lotfollahi, M. et al. Mapping single-cell data to reference atlases by transfer learning. *Nat. Biotechnol.* **40**, 121–130 (2022).
19. Hao, Y. et al. Integrated analysis of multimodal single-cell data. *Cell* **184**, 3573–3587.e29 (2021).
20. Kang, J. B. et al. Efficient and precise single-cell reference atlas mapping with Symphony. *Nat. Commun.* **12**, 1–21 (2021).
21. Chen, T., Kornblith, S., Norouzi, M. & Hinton, G. A simple framework for contrastive learning of visual representations in *International Conference on Machine Learning* 1597–1607 (PMLR, 2020).
22. He, K., Fan, H., Wu, Y., Xie, S. & Girshick, R. Momentum contrast for unsupervised visual representation learning in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* 9729–9738 (IEEE, 2020).
23. Wang, T. & Isola, P. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning* 9929–9939 (PMLR, 2020).
24. Chen, T., Kornblith, S., Swersky, K, Norouzi, M. & Hinton, G. Big self-supervised models are strong semi-supervised learners. Preprint at https://arXiv.org/quant-ph/2006.10029 (2020).
25. Hinton, G., Vinyals, O. & Dean, J. Distilling the knowledge in a neural network. Preprint at https://arXiv.org/quant-ph/1503.02531 (2015).
26. Anil, R. et al. Large scale distributed neural network training through online distillation. Preprint at https://arXiv.org/quant-ph/1804.03235 (2018).
27. Xie, Q., Luong, M. T., Hovy, E. & Le, Q. V. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* 10687–10698 (IEEE, 2020).
28. Fang, Z. et al. SEED: self-supervised distillation for visual representation. In *International Conference on Learning Representations* (2021).
29. Caron, M. et al. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* 9650–9660 (IEEE, 2021).
30. Vaswani, A. et al. Attention is all you need. In *Advances in Neural Information Processing Systems* 5998–6008 (2017).
31. Yang, Z. et al. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* 1480–1489 (Association for Computational Linguistics, 2016).
32. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. & Salakhutdinov, R. Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **15**, 1929–1958 (2014).

33. Gao, T., Yao, X. & Chen, D. SimCSE: Simple Contrastive Learning of Sentence Embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 6894–6910, Online and Punta Cana, Dominican Republic (Association for Computational Linguistics, 2021).
34. Chang, W. G., You, T., Seo, S., Kwak, S. & Han, B. Domain-specific batch normalization for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* 7354–7362 (IEEE, 2019).
35. Ding, J. et al. Systematic comparison of single-cell and single-nucleus RNA-sequencing methods. *Nat. Biotechnol.* **38**, 737–746 (2020).
36. Li, C. et al. SciBet as a portable and fast single cell type identifier. *Nat. Commun.* **11**, 1–8 (2020).
37. Cao, Z. J., Wei, L., Lu, S., Yang, D. C. & Gao, G. Searching large-scale scRNA-seq databases via unbiased cell embedding with Cell BLAST. *Nat. Commun.* **11**, 1–13 (2020).
38. Aran, D. et al. Reference-based analysis of lung single-cell sequencing reveals a transitional profibrotic macrophage. *Nat. Immunol.* **20**, 163–172 (2019).
39. Wagner, F. & Yanai, I. Moana: a robust and scalable cell type classification framework for single-cell RNA-Seq data. Preprint at https://arXiv.org/quant-ph/2018:456129 (2018).
40. Brbić, M. et al. MARS: discovering novel cell types across heterogeneous single-cell experiments. *Nat. Methods* **17**, 1200–1206 (2020).
41. Zappia, L., Phipson, B. & Oshlack, A. Splatter: simulation of single-cell RNA sequencing data. *Genome Biol* **18**, 1–15 (2017).
42. Park, J. E. et al. A cell atlas of human thymic development defines T cell repertoire formation. *Science* **367**, eaay3224 (2020).
43. Xie, Q., Dai, Z., Hovy, E., Luong, T. & Le, Q. V. Unsupervised data augmentation for consistency training. *Adv. Neural Inf. Process. Syst.* **33**, 6256–6268 (2020).
44. Cortal, A., Martignetti, L., Six, E. & Rausell, A. Gene signature extraction and cell identity recognition at the single-cell level with Cell-ID. *Nat. Biotechnol.* **39**, 1–8 (2021).
45. Tian, T., Wan, J., Song, Q. & Wei, Z. Clustering single-cell RNA-seq data with a model-based deep learning approach. *Nat. Mach. Intell.* **1**, 191–198 (2019).
46. Xie, J., Girshick, R. & Farhadi, A. Unsupervised deep embedding for clustering analysis. In *International Conference on Machine Learning* 478–487 (PMLR, 2016).
47. Wang, J. et al. scGNN is a novel graph neural network framework for single-cell RNA-Seq analyses. *Nat. Commun.* **12**, 1–11 (2021).
48. Xu, Y. et al. scIGANs: single-cell RNA-seq imputation using generative adversarial networks. *Nucleic Acids Res.* **48**, e85–e85 (2020).
49. Traag, V. A., Waltman, L. & van Eck, N. J. From Louvain to Leiden: guaranteeing well-connected communities. *Sci Rep.* **9**, 1–12 (2019).
50. Klein, A. M. et al. Droplet barcoding for single-cell transcriptomics applied to embryonic stem cells. *Cell* **161**, 1187–1201 (2015).
51. Wang, X. et al. BREM-SC: a bayesian random effects mixture model for joint clustering single cell multi-omics data. *Nucleic Acids Res.* **48**, 5814–5824 (2020).
52. Kim, H. J., Lin, Y., Geddes, T. A., Yang, J. Y. H. & Yang, P. CiteFuse enables multi-modal analysis of CITE-seq data. *Bioinformatics* **36**, 4137–4143 (2020).
53. Gayoso, A. et al. Joint probabilistic modeling of single-cell multi-omic data with totalVI. *Nat. Methods* **18**, 272–282 (2021).
54. Baron, M. et al. A single-cell transcriptomic map of the human and mouse pancreas reveals inter- and intra-cell population structure. *Cell Syst.* **3**, 346–360 (2016).
55. Segerstolpe, Å. et al. Single-cell transcriptome profiling of human pancreatic islets in health and type 2 diabetes. *Cell Metab.* **24**, 593–607 (2016).
56. Lawlor, N. et al. Single-cell transcriptomes identify human islet cell signatures and reveal cell-type-specific expression changes in type 2 diabetes. *Genome Res.* **27**, 208–222 (2017).
57. Grün, D. et al. De novo prediction of stem cell identity using single-cell transcriptome data. *Cell Stem Cell* **19**, 266–277 (2016).
58. Muraro, M. J. et al. A single-cell transcriptome atlas of the human pancreas. *Cell Syst.* **3**, 385–394 (2016).
59. Büttner, M., Miao, Z., Wolf, F. A., Teichmann, S. A. & Theis, F. J. A test metric for assessing single-cell RNA-seq batch correction. *Nat. Methods* **16**, 43–49 (2019).
60. Batool, F. & Hennig, C. Clustering with the average silhouette width. *Comput. Stat. Data Anal.* **158**, 107190 (2021).
61. Luecken, M. D. et al. Benchmarking atlas-level data integration in single-cell genomics. *Nat. Methods* **19**, 41–50 (2022).
62. Ren, X. et al. COVID-19 immune features revealed by a large-scale single-cell transcriptome atlas. *Cell* **184**, 1895–1913 (2021).
63. Zhu, L. et al. Single-cell sequencing of peripheral mononuclear cells reveals distinct immune response landscapes of COVID-19 and influenza patients. *Immunity* **53**, 685–696 (2020).
64. Schulte-Schrepping, J. et al. Severe COVID-19 is marked by a dysregulated myeloid cell compartment. *Cell* **182**, 1419–1440 (2020).
65. Su, Y. et al. Multi-omics resolves a sharp disease-state shift between mild and moderate COVID-19. *Cell* **183**, 1479–1495 (2020).
66. Mathew, D. et al. Deep immune profiling of COVID-19 patients reveals distinct immunotypes with therapeutic implications. *Science* **369**, eabc8511 (2020).
67. Raju, S., Kometani, K., Kurosaki, T., Shaw, A. S. & Egawa, T. The adaptor molecule CD2AP in CD4 T cells modulates differentiation of follicular helper T cells during chronic LCMV infection. *PLoS Pathog.* **14**, e1007053 (2018).
68. Tan, M. et al. Immunopathological characteristics of coronavirus disease 2019 cases in Guangzhou, China. *Immunology* **160**, 261–268 (2020).
69. Maucourant, C. et al. Natural killer cell immunotypes related to COVID-19 disease severity. *Sci. Immunol.* **5**, eabd6832 (2020).
70. Ioffe, S. & Szegedy, C. Batch normalization: accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning* 448–456 (PMLR, 2015).
71. *HOM_MouseHumanSequence* (Mouse Genome Infromatics, accessed 15 August 2020); http://www.informatics.jax.org/downloads/reports/HOM_MouseHumanSequence.rpt.
72. Giguere, C. et al. SCSIM: jointly simulating correlated single-cell and bulk next-generation DNA sequencing data. *BMC Bioinform.* **21**, 215 (2020).
73. Yang, Y. *Source Data of Concerto* (FigShare, 2022); https://doi.org/10.6084/m9.figshare.19351766

Acknowledgements

This research is supported by National Key R&D Program of China (grant no. 2021YFF1200105).

Author contributions

M.Y. conceived the problem and designed detailed study. H.M.Y., M.N. and J.L. allocated related resources and gave relevant advice. J.W. and F.M. supervised the work and provided strategic guidance. Y.Y. performed bioinformatics analysis. M.Y. and C.X. performed algorithm design and deep learning experiments. M.Y. wrote the manuscript, other authors made modifications.

Competing interests

J.W. and F.M. declare stock holdings in MGI, BGI-Shenzhen. The remaining authors declare no competing interests.

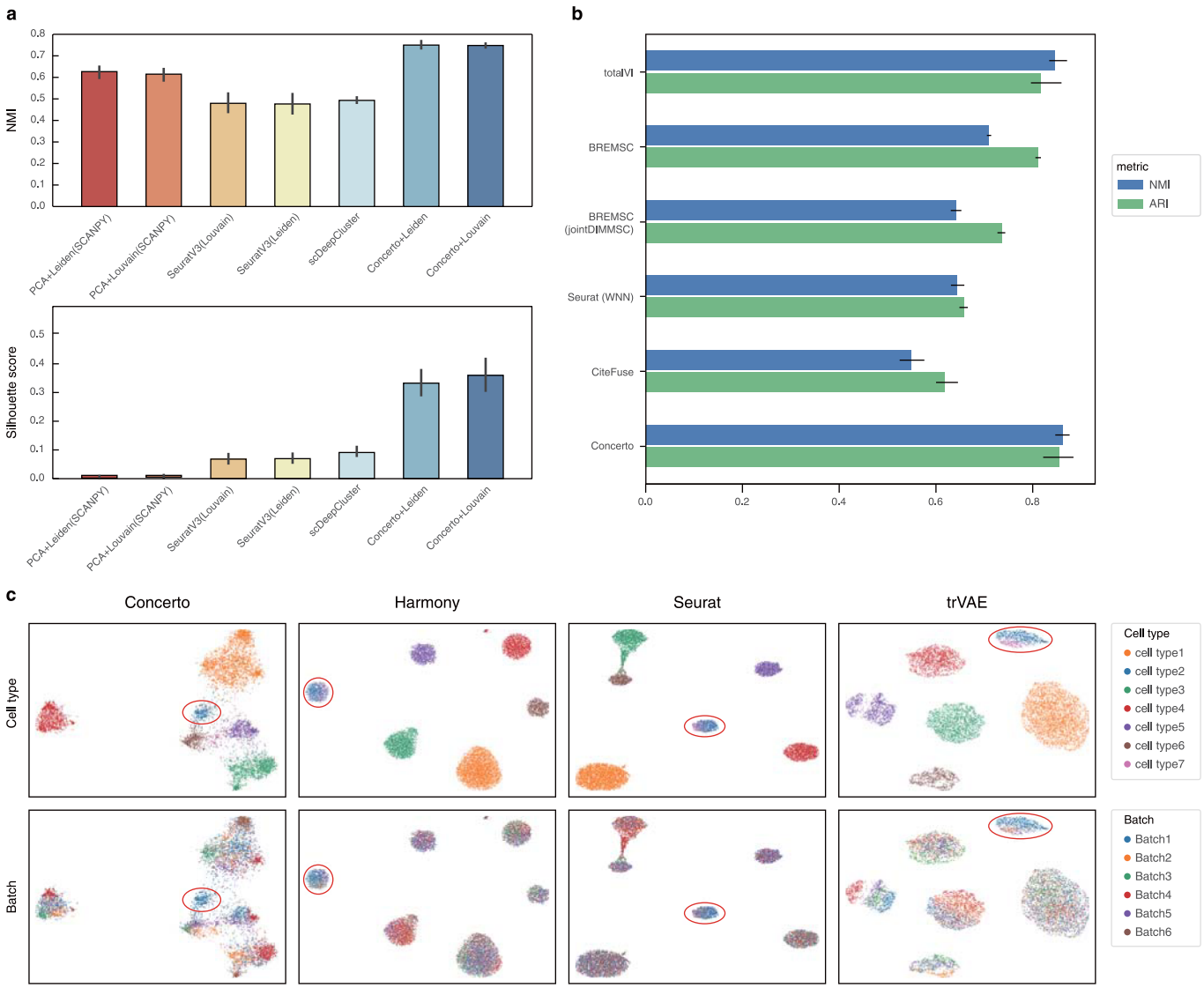
Additional information

Extended data is available for this paper at https://doi.org/10.1038/s42256-022-00518-z.

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s42256-022-00518-z.

Correspondence and requests for materials should be addressed to Meng Yang, Feng Mu or Jian Wang.

Peer review information *Nature Machine Intelligence* thanks the anonymous reviewers for their contribution to the peer review of this work.



Extended Data Fig. 1 | (a) Clustering performance benchmark measured by mean normalization mutual information (NMI) and Silhouette score on PBMC45K dataset (10X-v2, v3 only, n = 11,377 cells) across 5 different resolutions (resolution = 0.1,0.2,0.3,0.4,0.6 for Louvain or Leiden, k = 7,8,9,10,11 for scDeepCluster). (b) Benchmarking Concerto's clustering performance against BREMSC, jointDIMMSC, CiteFuse, totalVI and Seurat (WNN) measured by mean NMI and ARI on PBMC160k dataset (RNA and Protein) across 5 different resolutions. Leiden clustering for Concerto and totalVI, spectral clustering as default for CiteFuse and smart local moving (SLM) algorithm for Seurat WNN clustering. CiteFuse's scores are mean scores of 8 samples, since "out of memory" error (more than 500 G) occurs when all 160 k cells are used as input. (c) Over-correction analysis on simulated dataset via removing all cell type-2-cells except in Batch1. Cell type-2-cells are coloured in blue and circled in red on UMAP plots.

nature portfolio

Corresponding author(s): Meng Yang, Feng Mu, Jian Wang

Last updated by author(s): May 30, 2022

Reporting Summary

Nature Portfolio wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Portfolio policies, see our Editorial Policies and the Editorial Policy Checklist.

Statistics

For all statistical analyses, confirm that the following items are present in the figure legend, table legend, main text, or Methods section.

n/a	Confirmed
☐	☒ The exact sample size (*n*) for each experimental group/condition, given as a discrete number and unit of measurement
☐	☒ A statement on whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
☐	☒ The statistical test(s) used AND whether they are one- or two-sided *Only common tests should be described solely by name; describe more complex techniques in the Methods section.*
☒	☐ A description of all covariates tested
☐	☒ A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons
☐	☒ A full description of the statistical parameters including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals)
☐	☒ For null hypothesis testing, the test statistic (e.g. *F*, *t*, *r*) with confidence intervals, effect sizes, degrees of freedom and *P* value noted *Give P values as exact values whenever suitable.*
☒	☐ For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings
☒	☐ For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes
☐	☒ Estimates of effect sizes (e.g. Cohen's *d*, Pearson's *r*), indicating how they were calculated

Our web collection on statistics for biologists contains articles on many of the points above.

Software and code

Policy information about availability of computer code

Data collection	Simulated dataset is generated by open source Splatter R package and other datasets are all downloaded.
Data analysis	Code link in Code Ocean platform (https://codeocean.com/capsule/0526514/tree) and github (https://github.com/melobio/Concerto-reproducibility). We used following open source R packages: Seurat V3(v3.9.9), Seurat V4(v4.0.0), Seurat WNN(v4.1.0), SingleR(v1.0.0), SciBet, Harmony(v0.1), Symphony(v0.1), BREMSC(v0.2.0), CiteFuse(v1.2.1), scater, scsim. We used following open source Python packages: Scanpy(v1.7.1), Moana, CellBLAST(v0.3.8), MARS, totalVI, scDeepCluster, scArches(v0.3), tensorflow-gpu(v2.0.0b1), numpy(v1.19.5), pandas(v1.1.5), scipy(v1.5.4), scikit-learn(v0.24.2), argparse(v1.4.0), loompy(v3.0.6), openTSNE(v0.4.4), umap-learn(v0.4.6), seaborn(v0.11.2), h5py(v2.10.0).

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors and reviewers. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Portfolio guidelines for submitting code & software for further information.

Data

Policy information about availability of data

All manuscripts must include a data availability statement. This statement should provide the following information, where applicable:

- Accession codes, unique identifiers, or web links for publicly available datasets
- A description of any restrictions on data availability
- For clinical datasets or third party data, please ensure that the statement adheres to our policy

We use Splat simulation method in the Splatter package (version 1.10.0) to generate the simulated dataset, following author's tutorial (https://github.com/theislab/scib-reproducibility/tree/main/notebooks/data_preprocessing/simulations).

The published datasets analyzed in this paper are available in raw form through their original authors (see details in the Data availability).
PBMC45K is downloaded from https://singlecell.broadinstitute.org/single cell/study/SCP424/single-cell-comparison-pbmc-data#study-summary;
Thymus atlas is downloaded from https://zenodo.org/record/3572422#.XyjUKRMzZwc;
Tabula Muris Senis is downloaded from http://snap.stanford.edu/mars/data/tms-facs-mars.tar.gz;
HP datasets are downloaded from GSE81076, GSE85241, GSE86469, E-MTAB-5061, GSE84133;
PBMC160K is downloaded from https://atlas.fredhutch.org/data/nygc/multimodal/pbmc_multimodal.h5seurat;
Schulte-Schrepping et al. covid dataset is downloaded from EGAS00001004571;
Zhu et al. is downloaded from CNSA at https://db.cngb.org/cnsa;
Ren et al. is downloaded from GSE158055.

Human research participants

Policy information about studies involving human research participants and Sex and Gender in Research.

Reporting on sex and gender	n/a
Population characteristics	n/a
Recruitment	n/a
Ethics oversight	n/a

Note that full information on the approval of the study protocol must also be provided in the manuscript.

Field-specific reporting

Please select the one below that is the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

☒ Life sciences ☐ Behavioural & social sciences ☐ Ecological, evolutionary & environmental sciences

For a reference copy of the document with all sections, see nature.com/documents/nr-reporting-summary-flat.pdf

Life sciences study design

All studies must disclose on these points even when the disclosure is negative.

Sample size	No experiments in this study.
Data exclusions	No experiments in this study.
Replication	No experiments in this study.
Randomization	No experiments in this study.
Blinding	No experiments in this study.

Reporting for specific materials, systems and methods

We require information from authors about some types of materials, experimental systems and methods used in many studies. Here, indicate whether each material, system or method listed is relevant to your study. If you are not sure if a list item applies to your research, read the appropriate section before selecting a response.

Materials & experimental systems

n/a	Involved in the study
☒	☐ Antibodies
☒	☐ Eukaryotic cell lines
☒	☐ Palaeontology and archaeology
☒	☐ Animals and other organisms
☒	☐ Clinical data
☒	☐ Dual use research of concern

Methods

n/a	Involved in the study
☒	☐ ChIP-seq
☒	☐ Flow cytometry
☒	☐ MRI-based neuroimaging