OXFORD

Briefings in Bioinformatics, 2024, **25(2)**, 1–19
https://doi.org/10.1093/bib/bbae039
Problem Solving Protocol

scEVOLVE: cell-type incremental annotation without forgetting for single-cell RNA-seq data

Yuyao Zhai, Liang Chen and Minghua Deng

Corresponding author. Mingua Deng, School of Mathematical Sciences, Peking University, Beijing 100871, China; Center for Quantitative Biology, Peking University, Beijing 100871, China; Center for Statistical Science, Peking University, Beijing 100871, China. Tel.: 13522856599. Email: dengmh@pku.edu.cn

Abstract

The evolution in single-cell RNA sequencing (scRNA-seq) technology has opened a new avenue for researchers to inspect cellular heterogeneity with single-cell precision. One crucial aspect of this technology is cell-type annotation, which is fundamental for any subsequent analysis in single-cell data mining. Recently, the scientific community has seen a surge in the development of automatic annotation methods aimed at this task. However, these methods generally operate at a steady-state total cell-type capacity, significantly restricting the cell annotation systems'capacity for continuous knowledge acquisition. Furthermore, creating a unified scRNA-seq annotation system remains challenged by the need to progressively expand its understanding of ever-increasing cell-type concepts derived from a continuous data stream. In response to these challenges, this paper presents a novel and challenging setting for annotation, namely cell-type incremental annotation. This concept is designed to perpetually enhance cell-type knowledge, gleaned from continuously incoming data. This task encounters difficulty with data stream samples that can only be observed once, leading to catastrophic forgetting. To address this problem, we introduce our breakthrough methodology termed scEVOLVE, an incremental annotation method. This innovative approach is built upon the methodology of contrastive sample replay combined with the fundamental principle of partition confidence maximization. Specifically, we initially retain and replay sections of the old data in each subsequent training phase, then establish a unique prototypical learning objective to mitigate the cell-type imbalance problem, as an alternative to using cross-entropy. To effectively emulate a model that trains concurrently with complete data, we introduce a cell-type decorrelation strategy that efficiently scatters feature representations of each cell type uniformly. We constructed the scEVOLVE framework with simplicity and ease of integration into most deep softmax-based single-cell annotation methods. Thorough experiments conducted on a range of meticulously constructed benchmarks consistently prove that our methodology can incrementally learn numerous cell types over an extended period, outperforming other strategies that fail quickly. As far as our knowledge extends, this is the first attempt to propose and formulate an end-to-end algorithm framework to address this new, practical task. Additionally, scEVOLVE, coded in Python using the Pytorch machine-learning library, is freely accessible at https://github.com/aimeeyaoyao/scEVOLVE.

Keywords: scRNA-seq data; cell-type incremental annotation; contrastive sample replay; cell-type decorrelation

INTRODUCTION

Single-cell RNA sequencing (scRNA-seq) technologies enable the profiling of gene expressions in millions of individual cells, offering unparalleled insights into the intricate cellular ecosystem [1, 2]. Crucial to analyzing scRNA-seq data is cell annotation since numerous downstream explorations hinge on the identification of specific cell types [3, 4]. A conventional single-cell annotation pipeline typically commences with the segmentation of cell data into various clusters, proceeds to identify each group's marker genes by differential expression analysis and culminates with group annotation through gene ontology knowledge. For instance, scCatch assigns annotation categories to cells based on canonical cell-type marker genes through a scoring approach grounded on evidence [5]. Nonetheless, this methodology necessitates manual verification of marker genes, entailing the review of extensive biological literature [6, 7]. Consequently, this process could pose a considerable challenge to researchers within nonspecialized fields [8].

As the availability of extensive, well-labelled reference data expands, numerous automated annotation strategies, based on projection and classification, have been emerging [9, 10]. For example, scmap utilizes new cells, projects them onto reference data and measures their similarity to known cluster centroids within the reference data to carry out annotation [11]. Seurat, conversely, predicts the supervised targets on the data by identifying pairs of anchor cells and subsequently building shared neighbor networks between the reference and target data [12]. Capitalizing on the advanced capabilities of deep learning in feature representation, studies have begun to employ nonlinear neural networks to facilitate cell classification. A case in point is scANVI, a generative model founded on a deep variational autoencoder-based scRNA-seq semi-supervised framework, developed with the aim of leveraging any available cell state annotations [13]. MARS adopts meta-learning in such a manner that identical cell types will share similar features, while any discrepancies in cell type will result in quite distinct features [14]. On a similar note, scArches employs domain adaptation and parameter

Yuyao Zhai is a doctoral candidate at the School of Mathematical Sciences, Peking University.
Liang Chen is a senior researcher at Huawei.
Minghua Deng is a professor at the School of Mathematical Sciences, Peking University.
Received: October 24, 2023. **Revised:** January 3, 2024. **Accepted:** January 9, 2024

optimization techniques to construct references and contextualize target data [15]. In conclusion, the majority of current annotation methods operate within the boundaries of established cell-type knowledge capacity, which, unfortunately, restricts their ability to discover new cell types from an influx of data. This limitation poses significant challenges to their applicability in real-world situations.

The emerging prominence of incremental learning has drawn scholarly interest in models capable of an ongoing learning process. This involves accommodating new tasks through the assimilation of fresh data while retaining previously acquired knowledge [16, 17]. A significant challenge posed by incremental learning involves averting catastrophic forgetting, a scenario where the performance on former tasks or domains suffers a significant decline over time upon the introduction of new ones [18, 19]. This challenge emanates from a broader issue in neural networks referred to as the stability-plasticity dilemma. Here, plasticity correlates with the capacity to incorporate new knowledge, while stability means retaining prior knowledge during encoding. Incremental learning research largely falls into three categories, differentiated by how task-specific information is stored and applied in the sequential learning process. These categories include replay methods [20, 21], regularization-based techniques [22, 23] and parameter isolation methods [24, 25]. For an in-depth overview of related studies, refer to the existing literature [26, 27].

Drawing upon incremental learning theories, it is rational to infer that the scRNA-seq annotation system operates on an inherently incremental basis, where it consistently integrates new cell heterogeneity information while retaining existing knowledge. To illustrate, researchers examining pancreatic tissue can discover numerous new cell types without losing their understanding of the cell types in intestinal and stomach tissues. Contrarily, the majority of annotation systems operate within a batch setting, implying they require prior knowledge of all cell types and simultaneous accessibility to all types'training data [28–30]. Notwithstanding, the sphere of scRNA-seq data analysis is progressively converging with artificial intelligence, demanding more adaptable approaches to managing the vast-scale and dynamic nature of practical cell annotation scenarios. Fundamentally, a solitary cell annotation system should possess the capability to progressively learn about novel cell types predicated on the availability of their training data. This system modality is referred to as incremental cell-type annotation.

To be classified as an incremental cell-type annotation algorithm, it should meet several formal criteria. Firstly, it should be capable of being trained from a continuous data stream where cells from varying cell types appear at different intervals. Secondly, it should deliver a competitive multi-class classifier for the cell types identified up to that point. Lastly, it should adhere to a stringent set of computational specifications and memory usage, or at least demonstrate a marginal increase in proportion to the number of observed cell types thus far. The initial two criteria underscore the practical applicability of these incremental annotation tasks in classifying cell types. The final specification rejects any simple algorithms that might, for instance, retain all training samples and employ a standard multi-class classifier whenever a new dataset is introduced [14, 31].

Despite significant advancements in automatic scRNA-seq annotation methods over recent years, a fully satisfactory cell-type incremental annotation algorithm remains elusive. Most current methodologies fall short of requirements 1 or 2, as they solely manage a predetermined number of cell types or necessitate simultaneous availability of all training data. Training classifiers from incrementally obtained cell-type data streams, such as through the application of stochastic gradient descent optimization, offers a potential solution. Yet, this approach often leads to a rapid diminution in annotation accuracy, a phenomenon we refer to as catastrophic forgetting in incremental learning. Notwithstanding the few techniques currently meeting the aforementioned criteria, their utility remains largely confined to scenarios with fixed data representation in the single-cell domain. Consequently, these techniques cannot be expanded to incremental architectures, which simultaneously learn continual classifiers and features, thus leaving them outdated in relation to annotation accuracy.

This paper introduces scEVOLVE, a new framework for incremental annotation of cell types that can seamlessly integrate new data without disregarding established knowledge. The framework advances from the standpoint of contrastive sample replay and cell-type decorrelation. To offset the risk of erasing prior cell-type information with the introduction of new data, scEVOLVE employs an episodic memory buffer. This feature stores a sampling of prior stage data to then be used in conjunction with the current stage. The selection of sample examples applies a protocol rooted in nearest prototype classification certainty; this protocol allows for an effective reduction of duplicate samples throughout the runtime while ensuring high proximity to the cell-type prototype. In the next stages, the framework could replay previous samples. However, conventional softmax-based prototype classification may contribute to bias issues resulting from cell-type imbalance. This can lead to the misplacement of many existing cell-type samples into new categories. To counter this, scEVOLVE employs a contrastive replay method. No longer relying on sample-to-anchor pairs, this method utilizes prototype-to-anchor pairs in the contrastive-based loss, strictly using prototypes that align with the cell types present in the same batch. The incremental model of scEVOLVE is carefully designed to mirror the oracle model in every phase. The goal is to make the full data-trained performance of the oracle model the ultimate target. To get there, we impose uniform scattering across cell-type data representations in each phase, emulating the oracle model representations. Partition confidence maximization drives the cell-type decorrelation regularization, making this model implementation specific.

To judge scEVOLVE's efficacy, we've selected several large-scale scRNA-seq datasets and designed an exhaustive set of cell-type incremental annotation benchmarks. A multitude of experiments using these benchmarks reveals scEVOLVE's power to mitigate the catastrophic forgetting problem while allowing incremental cell-type learning over extended timeframes. Most importantly, scEVOLVE manages the forgetting problem by balancing gradient propagation and regularizing the scatter of embedding representations. Further investigation verifies the effectiveness of every module. Lastly, we believe our method can be comfortably incorporated into a variety of deep softmax-based, single-cell annotation techniques, making it a worthy candidate for widespread integration.

METHOD

We commence our discussion with the establishment of the problem context and the associated notations. In order to closely align with the practical scenario, we understand that scRNA-seq data are stored in a data flow and subsequently analyzed by the model in a cell type-incremental format. Specific sample sets illustrated by $\{\mathcal{X}^1, \mathcal{X}^2, ...\}$ and corresponding label sets denoted by $\{\mathcal{Y}^1, \mathcal{Y}^2, ...\}$ could either originate from the same or disparate scRNA-seq datasets. To enhance the efficacy of model learning,

we segregate the data stream into a series of learning tasks, represented as $\mathcal{D} = \{\mathcal{D}^i\}_{i=1}$. Here, $\mathcal{D}^i = \{\mathcal{X}^i \times \mathcal{Y}^i, \mathcal{C}^i\}$ includes sample sets $\mathcal{X}^i$, the corresponding label sets $\mathcal{Y}^i$, as well as task-specific cell types $\mathcal{C}^i$. Each sample set, $\mathcal{X}^i$, is hypothetically divided at random into a training dataset and a testing dataset. These can be denoted, respectively, as $\mathcal{X}^{ir} = \{(x_j^{ir}, y_j^{ir})_{j=1}^{n_{ir}}\}$ and $\mathcal{X}^{it} = \{(x_j^{it}, y_j^{it})_{j=1}^{n_{it}}\}$. Within these, the labels within the training dataset, y_j^{ir}, are known; however, the labels within the test set, y_j^{it}, are yet to be predicted. Furthermore, the label relationship among any two given datasets within the data flow may exhibit partial overlap. This is represented by the conditions $\mathcal{C}^i \cap \mathcal{C}^j \neq \varnothing$, $\mathcal{C}^i \setminus \mathcal{C}^j \neq \varnothing$, and $\mathcal{C}^j \setminus \mathcal{C}^i \neq \varnothing$.

While many annotation methods have performed satisfactorily under constant total cell-type scenarios in recent years, the introduction of a new, unrelated dataset can trigger what is often referred to as catastrophic forgetting issues. This phenomenon is characterized by the erasure of previously learned information about old cell types. Yet, our natural expectation is a continual evolution of the annotation system over time, accommodating an increasing repertoire of cell types. As such, our aim is to formulate a universal algorithm that supports incremental learning of new cell types, reflecting the ever-changing demands of real-world cell annotation. With this algorithm, when cells from new types need annotating, the model will adapt, recognizing their gene expression patterns without neglecting the knowledge previously acquired. Repeated training of all cells from pre-existing cell types will be eliminated, resulting in a significant rise in the efficiency of the cell annotation system. Moreover, our task is executed within an inductive learning framework, a divergence from the transductive learning approach adopted by most prior domain adaptation-based single-cell annotation methods [32–35]. Such a setup considers the practical implications of deploying the cell annotation system in real-world environments without incurring additional costs associated with adjusting the test data.

Basic network construction of model

Our model is comprised of a Zero-Inflated Negative Binomial-oriented denoising autoencoder [36], possessing a reconstruction loss denoted as L_{zinb}, and a prototype-parameterized classifier symbolized as Φ (refer to Figure 1). Comprehensive details can be found in the Supplementary Information (SI). The classifier, Φ, which is appended subsequent to the encoder labeled $z = f_e(x)$, can be delineated as $\Phi = \langle z, V\rangle/\tau$. Herein, $V = [v_1, \ldots]^T$ houses the trainable cell-type prototypes learned thus far, $\langle\cdot,\cdot\rangle$ signifies the cosine similarity, and τ is a scaling factor. Subsequently, the similarity vector is normalized by the softmax function to derive the probability of a sample's alignment with different prototypes. As an example, consider the s-stage of training, where all learned cell types are abbreviated as $\mathcal{C}_{1:s}^r = \bigcup_{i=1}^{s} \mathcal{C}^{ir}$. The likelihood that sample x_i pertains to the cell type c is represented in

$$p_{sc}^i = \frac{exp(\langle f_e^s(x_i), v_c^s\rangle/\tau)}{\sum_{l\in\mathcal{C}_{1:s}^r} exp(\langle f_e^s(x_i), v_l^s\rangle/\tau)}. \tag{1}$$

It is pertinent to note that as observed cell types increase, so does the number of classification heads. In the sth training phase, the model has access solely to the $\mathcal{D}^s$ data related to that phase, and each sample can only be observed once. The classification objective utilizes the cross-entropy function, expressed in

$$L_{ind}^s = E_{(x,y)\sim\mathcal{D}^s}\left[-log\left(\frac{exp(\langle f_e^s(x), v_y^s\rangle/\tau)}{\sum_{l\in\mathcal{C}_{1:s}^r} exp(\langle f_e^s(x), v_l^s\rangle/\tau)}\right)\right]. \tag{2}$$

From the provided formula, it is clear that L_{ind}^s trains the model solely using current stage data, a method frequently referred to as individual training. This approach does not aim to prevent forgetting. Contrastingly, joint learning represents the other extremity, where the model is trained utilizing all samples from both prior $s-1$ stages, and the current sth stage. Joint learning's cross-entropy loss is represented in

$$L_{joi}^s = E_{(x,y)\sim\bigcup_{i=1}^{s}\mathcal{D}^i}\left[-log\left(\frac{exp(\langle f_e^s(x), v_y^s\rangle/\tau)}{\sum_{l\in\mathcal{C}_{1:s}^r} exp(\langle f_e^s(x), v_l^s\rangle/\tau)}\right)\right]. \tag{3}$$

As such, the performance of joint learning can be posited as the upper boundary of our task.

Exemplar set construction and management

Firstly, this study investigates the underlying causes of catastrophic forgetting during cell-type incremental annotation, examining it through the lens of gradient propagation. The calculation of the gradient for a single data point or sample represented as x with its corresponding label y at stage s can be determined as depicted in

$$\frac{\partial L_{ind}^s}{\partial V^s} = \begin{cases} f_e^s(x)(p_{sy}-1), & c = y \\ f_e^s(x)p_{sc}, & c \neq y, \end{cases} \tag{4}$$

Here, the symbol c denotes a specific cell type within the existing label set. Drawing from this equation, it can be inferred that the dimension of the gradient vector corresponding to y experiences a decrease by $p_{sy}-1 < 0$, while the alternate dimension sees an increase by $p_{sc} > 0$. Applying the chain rule in conjunction with this, the process of model updating will furnish a positive gradient for the prototype of the cell type represented as y with $v_y^s = v_y^s - \eta f_e^s(x)(p_{sy}-1)$, and disperse a negative gradient to the rest of the prototypes with $v_c^s = v_c^s - \eta f_e^s(x)p_{sc}$. This implies that upon adjusting the model by directly optimizing the loss function, depicted as L_{ind}^s, it can be inferred that the learning process associated with new cell types governs gradient propagation. This results in the occurrence of catastrophic forgetting.

Based on the above analysis, a critical step in managing gradient propagation and averting the forgetting problem is the storage of previously experienced samples. However, this necessitates considering two prerequisites for storing samples. Commonly referred to as joint learning, the method of storing all training samples previously encountered necessitates considerable memory resources, particularly as the quantum of previously learned samples mounts. Conversely, a haphazard approach of storing samples oblivious of their cell type may result in certain cell types left without any stored samples—thus deleteriously affecting model performance. Given these two prerequisites, careful selection of a prior sample subset as representative samples stored in the memory buffer $\mathcal{M}$ to augment the current training set is essential. With stage s exemplifying, the observed cell types thus far can be represented as $\{\mathcal{C}^1, \mathcal{C}^2, \ldots, \mathcal{C}^{s-1}\}$, while the quantity of cell types newly introduced at stage i can be represented as $k_i = |\mathcal{C}^i \setminus (\bigcup_{k=1}^{i-1} \mathcal{C}^k \cap \mathcal{C}^i)|$. Subsequently, the representative sample sets at stage s, denoted $E_1^s, \ldots, E_{k_{1:s-1}}^s$, are dynamically contrived from the data stream with $k_{1:s-1} = \sum_{i=1}^{s-1} k_i$. The training set for the sth time period, denoted as $\mathcal{X}^{s*}$, can be articulated as $\mathcal{X}^{s*} = \mathcal{X}^s \cup \bigcup_{i=1}^{k_{1:s-1}} E_i^s$. To constrain memory requirements, hyperparameter m is deemed a fixed value, representing the quantity of representative samples

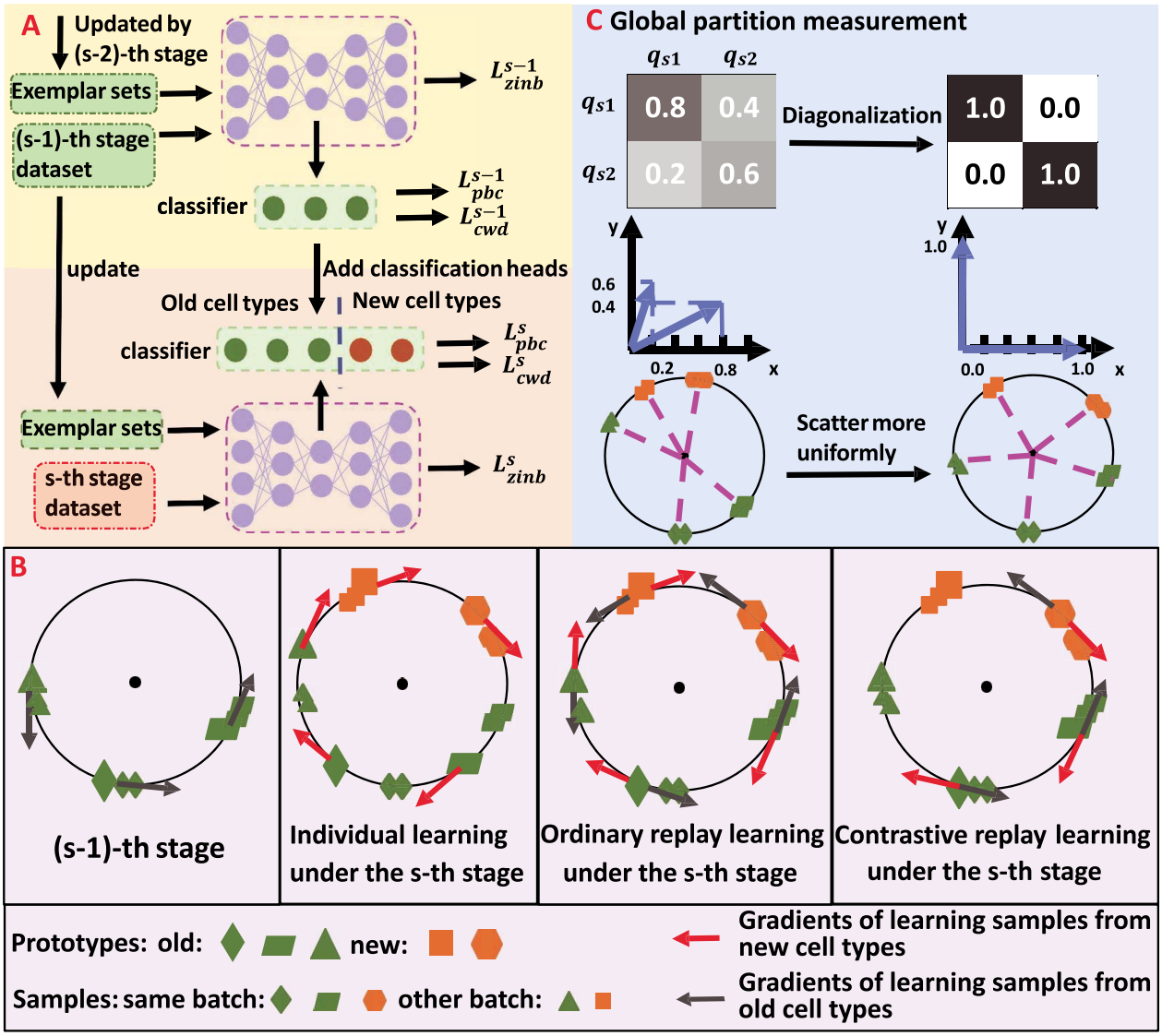

Figure 1. The following text provides an overview of scEVOLVE: (**A**) During the sth stage, the model updates the exemplar sets based on the dataset from the $s-1$th stage. This updated set, in conjunction with the sth stage dataset, forms the model's input. When new cell types emerge at the sth stage, matching new heads are incorporated into the classifier. The formulation $\mathcal{L}^s_{pbc}$ is devised to balance the gradient progression between existing and novel cell types, while the formula $\mathcal{L}^s_{cwd}$ enforces a uniform scattering of data representations pertaining to different cell types in the embedding space. (**B**) The text includes an illustration depicting the procedure of gradient propagation from individual samples to all prototypes under various learning strategies. Our approach more effectively manages the gradient propagation process, thereby enhancing the identification of both new and existing cell types. (**C**) A further illustration demonstrates the compression technique applied to the global partition measurement values, excluding the diagonal elements, facilitating the learning process to achieve the most confident and promising classification solution.

for each cell type. During the revolutionary selection process, the selected representatives are modelled to closely mimic the corresponding cell-type prototype. For a specific cell type $\mathcal{C}^{cr}$ at stage s, the similarity of each constituent sample x^{sr}_{ic} to its corresponding prototype v^s_c can be determined as $s_{ic} = \langle f^s_e(x^{sr}_{ic}), v^s_c\rangle/\tau$, allowing us to single out the top m samples with the highest similarity as representatives for that cell type. Furthermore, there's no requirement to reselect a representative set for previously learned cell types at the current stage. Once the representative sample set has been assembled, it simplifies the broad implementation of the loss function L^s_{ind}, as depicted in

$$L^s_{rep} = E_{(x,y)\sim\mathcal{D}^s\cup\mathcal{M}}\left[-log\left(\frac{exp(\langle f^s_e(x), v^s_y\rangle/\tau)}{\sum_{l\in\mathcal{C}^r_{1:s}} exp(\langle f^s_e(x), v^s_l\rangle/\tau)}\right)\right]. \quad (5)$$

Prototypical contrastive replay learning

While we establish a corresponding exemplar set for each distinct cell type, the issue of an imbalanced gradient propagation persists during batch training with L^s_{rep}. The feature extractor specifically focuses its attention on the attributes of newly introduced cell types. This results in the positioning of new and old cell samples in close proximity within the embedding space, thereby facilitating the categorization of samples into new cell clusters. As training progresses, the gradient corresponding to old cell types proves inadequate, given that each cell type's representative samples are fixed in the memory buffer. The newly introduced cell types then monopolize this process, rendering their respective samples highly distinguishable, while simultaneously causing those of the old cell types to become almost indiscernible. Therefore, efficiently governing the gradient flow between the old and new

cell types could significantly favor the mitigation of the forgetting problem.

Based on the memory buffer $\mathcal{M}$, a selective retraining of previous samples of old cell types can be used to regulate gradient propagation. This regulation can be accomplished by choosing particular anchor-to-prototype pairs and computing their classification objective. This strategy may curb the imbalance of gradient propagation by guiding samples back to the corresponding cell-type prototypes. However, this could potentially disturb the model's generalization capabilities because it may introduce a bias against acquiring new cell types. An alternative conceivable solution is to implement contrastive-based loss [37, 38], represented as

$$L_{con}^{s} = E_{(x,y)\sim\mathcal{D}^{s}\cup\mathcal{M}}\left[-log\left(\frac{exp(\langle f_{e}^{s}(x), f_{e}^{s}(x_{a})\rangle/\tau)}{\sum_{j\in J(x)} exp(\langle f_{e}^{s}(x), f_{e}^{s}(x_{j})\rangle/\tau)}\right)\right]. \quad (6)$$

The contrastive-based loss integrates current and previous samples into one batch and assesses the similarities of anchor-to-sample pairs. Here, $J(x)$ symbolizes the index set of samples, excluding anchor x in the same batch, whereas $a \in A(x)$ represents the set of samples with identical labels to the anchor x. Unlike softmax-based loss, the selected pairs do not depend on the number of cell types, they are instead related to the number of samples in a training batch. Consequently, its effectiveness is restricted by the dimensions of the memory buffer and batch size. In scenarios with only a few samples for replay, its performance may not be satisfactory. To tackle these issues, we suggest employing prototype-based contrastive learning. This approach could prevent the occurrence of catastrophic forgetting by combining the benefits of both prototype- and contrastive-based learning methods. More specifically, we substitute the samples of anchor-to-sample pairs with prototypes in contrastive-based loss. In the s-time period, for instance, our coupling-inspired prototype-based contrastive loss is defined as

$$L_{pbc}^{s} = E_{(x,y)\sim\mathcal{D}^{s}\cup\mathcal{M}}\left[-log\left(\frac{exp(\langle f_{e}^{s}(x), v_{y}^{s}\rangle/\tau)}{\sum_{l\in\mathcal{C}_{B}^{s}} exp(\langle f_{e}^{s}(x), v_{l}^{s}\rangle/\tau)}\right)\right], \quad (7)$$

where $\mathcal{C}_{B}^{s}$ represents the cell type indices in the current training batch at the sth stage, with the indices being potentially repeated. By leveraging prototypes, this loss can handle small sets of exemplars, thus eluding the limitation imposed by memory buffer size. Moreover, the substituted prototypes are derived only from the cell types present in the training batch, ensuring that propagation gradients originate solely from the learning of these cell types. In each learning iteration, only the new and old cell types in the current batch contribute to gradient propagation. The prototypes of old cell types, which are influenced by the negative gradient of new cell types, have the capacity to generate a positive gradient for conflict resolution and additionally alleviate the forgetting issue. In conclusion, the integration of contrastive learning and prototype learning can yield an optimal solution for incremental cell-type annotation.

Cell-type decorrelation learning

While we manage to mitigate catastrophic forgetting to a certain extent by calibrating gradient propagation across new and old cell categories, we contend that refining the embedding representations at each phase is paramount to enhancing the model's efficacy. All data can be concurrently processed in joint learning, ostensibly allowing the taught representations of every cell category to be evenly distributed within the embedding space. In stark contrast, a model trained with a limited number of cell categories tends to arrange the representations in a long, linear region, generating a more evenly dissected representation with an expanding number of training cell categories [39, 40]. Our preference is towards applying model regularization to yield scattered representations akin to those found in joint learning at every stage. The zenith of our task might be joint learning as it advocates for simultaneous data processing. We propose that if collectively learned representations of each cell type are uniformly spread within the embedding space, mimicking its representation could enhance performance at the present stage while simultaneously rendering the current stage representations more adaptable for incremental processing of new cell categories.

Stemming from our motivation, we introduce the concept of Global Partition Measurement (GPM). This concept is designed to distinctly separate samples of varying cell types following a learning model that identifies the most secure partitioning method. To further extend on this, consider an example where the stage of operation is s - in this context, the training data can be presented as $\mathcal{X}^{sr*} = \{(x_j^{sr}, y_j^{sr})_{j=1}^{n_{sr}}\} \cup \bigcup_{i=1}^{k_{1:s-1}} E_i^s$. Accordingly, we can then determine the prediction probability for each sample x_i at the sth stage, represented as $p_s^i = (p_{s1}^i, p_{s2}^i, ..., p_{sk_{1:s}}^i)^T \in \mathcal{R}^{k_{1:s}\times 1}$. Here, $k_{1:s}$, which equals $\sum_{i=1}^{s} k_i$, corresponds to the total number of cell types learned after the sth phase. From this, we can derive the prediction matrix for all n_{sr*} samples, as shown in

$$p_s = [p_s^1, p_s^2, ..., p_s^{n_{sr*}}] \in \mathcal{R}^{k_{1:s}\times n_{sr*}}. \quad (8)$$

For ease of interpretation, we reference the jth row of p_s in

$$q_{sj} = [p_{sj}^1, p_{sj}^2, ..., p_{sj}^{n_{sr*}}] \in \mathcal{R}^{1\times n_{sr*}}, j \in [1, 2, ..., k_{1:s}]. \quad (9)$$

Given that q_{sj} accumulates the probability values of all samples corresponding to the jth cell type, it can be termed the cell type assignment vector', illustrating the broader assignment statistics of that specific cell type. For ensuring a homogenous distribution of data across each cell type, we posit that any two vectors, q_{sa} and q_{sb}, should ideally be orthogonal, implying a cosine similarity of zero, as represented in

$$cos(q_{sa}, q_{sb}) = \frac{q_{sa}\cdot q_{sb}}{\|q_{sa}\|_2\|q_{sb}\|_2}, a, b \in [1, 2, ..., k_{1:s}]. \quad (10)$$

In accordance with the preceding analysis, we describe the GPM during the sth stage as the cosine similarity set amounting to all the pairs of cell-type assignment vectors, as shown in

$$G_s(a, b) = cos(q_{sa}, q_{sb}), a, b \in [1, 2, ..., k_{1:s}]. \quad (11)$$

Here, G_s manifests as a $k_{1:s} \times k_{1:s}$ matrix. Our objective revolves around reducing the GPM values, excluding diagonal elements and facilitating maximal separation amongst various cell types, thus ensuring a uniform dispersion in the embedding space. This training aim bolsters self-attention by viewing each cell type as a standalone data sample while diminishing attention between samples. Therefore, we apply the softmax operation to each cell type a, obtaining a probability measurement, as represented in

$$g_s(a, a') = \frac{exp(G_s(a, a'))}{\sum_{k=1}^{k_{1:s}} exp(G_s(a, k))}, a' \in [1, 2, ..., k_{1:s}]. \quad (12)$$

Subsequently, we administer the loss of cross-entropy, considering $g_s(a, a)$ as the model prediction probability aligned with a training sample's ground-truth label, illustrated in

$$L^s_{cwd} = \frac{1}{k_{1:s}} \sum_{a=1}^{k_{1:s}} -log(g_s(a, a)). \tag{13}$$

By formulating L^s_{cwd}, we can effectively curtail non-diagonal elements of the GPM matrix, $G_s(a, b)$, thereby ensuring a uniform representation of different cell types.

In conclusion, for the s-stage, we incorporate the data denoising loss L^s_{zinb} to provide the comprehensive training objective, represented as

$$L^s_{Overall} = L^s_{zinb} + \lambda L^s_{pbc} + \gamma L^s_{cwd}, \tag{14}$$

where λ and γ are loss weight hyperparameters.

PERFORMANCE EVALUATION

Dataset composition

To encompass various potential scenarios for incremental learning of cell types, we categorize our experiments into three distinct sections: intra-data, inter-tissue and inter-data. It should be noted that batch effects are present between training and testing data in the latter two categories. Each category includes a selection of large-scale, atlas-level datasets with a substantial imbalance in cell-type composition. Our choice for the intra-data category includes the datasets of Cao, featuring 16 cell types and 32 061 cells [41], Quake 10x with 36 cell types and 54 967 cells [42], Quake Smart-seq2 comprising 45 cell types and 44 807 cells [42], and Zeisel encompassing 21 cell types and 160 796 cells [43]. Unless otherwise specified, we divide their cell types into four exclusive subcategories accounting for the cell types available at each stage. The training and testing sets are then apportioned based on a 1:9 ratio, which results in a labeled ratio of 0.1. For the inter-tissue category, the Cao_2020 atlas provides us with four tissues [44]: Eye with 11 cell types and 51 836 cells, Intestine with 12 cell types and 51 650 cells, Pancreas with 13 cell types and 45 653 cells, and Stomach consisting of 10 cell types and 12 106 cells. A degree of overlap is observed between cell types in paired tissues. These tissues are trained in alternating alphabetical orders at various stages. Lastly, for the inter-data segment, we utilize four datasets from different tissues that have been sequenced on distinct platforms. Specifically, we use He with 11 cell types and 15 680 cells [45], Madissoon with 17 cell types and 57 020 cells [46], Stewart with 18 cell types and 26 628 cells [47], and Vento containing 17 cell types and 64 734 cells [48]. Again, these are alternately trained in alphabetical order. Additionally, a split labeled ratio of 0.1 is maintained for the training and testing data in both the second and third categories. For complete details regarding these datasets, please refer to the SI.

Evaluated baselines

Our methodology targets the resolution of catastrophic forgetting problems occurring in cell-type incremental annotation learning—a niche befitting to our approach due to an absence of established annotation baselines. In a bid to demonstrate the effectiveness of scEVOLVE, we juxtapose it against four distinct methodological categories. The preliminary category utilizes an individual training strategy. This strategy is based on the training of the model exclusively on the data from the current stage. The exemplified training loss for this strategy can be numerically defined as $L^s_{zinb} + \lambda L^s_{ind}$, with the performance levels serving as the fundamental baseline for our task. The second often-employed strategy is the joint learning approach, using all data gathered thus far for training the model. The loss encountered in joint learning could be quantified as $L^s_{zinb} + \lambda L^s_{joi}$, and the corresponding performance serves as the upper-efficiency boundary for our task. Thirdly, with an aim to illustrate the influence of sample replay, the current data, alongside the exemplars, employ the cross-entropy loss, represented as $L^s_{zinb} + \lambda L^s_{rep}$. In light of this, we substituted L^s_{rep} with L^s_{pbc}, which is represented as $L^s_{zinb} + \lambda L^s_{pbc}$, to monitor the improvements resulting from replay based on prototype-based contrastive methods. Subsequently, integrating cell-type decorrelation regularization into scEVOLVE alludes to an examination of how this principle could impact performance. To ensure that the modules put forth in our proposed methodology could be seamlessly integrated into existing single-cell annotation procedures, we selected three exemplary methods: scmap hinged on traditional statistical learning [11], scANVI leveraged on deep representation learning [13] and CIForm based on transformer structure [49]. Considering that scmap and scANVI do not rely on the deep softmax classification mechanism, only the sample replay module can be integrated into their implementational design. Contrarily, all three proposed modules are compatible with CIForm.

Evaluation metrics

We evaluate the performance of all methods by reporting their classification accuracy. The term Old Accuracy pertains to the methods'classification accuracy derived from the testing data from all preceding stages. New Accuracy signifies the methods'classification accuracy as based on the testing data culled from the current stage. Moreover, Overall Accuracy is the summative classification accuracy on the collective data set, which includes testing data from both previous and current stages. The accuracy values that we report represent the average value culled from three separate runs.

Implementation details

Our algorithm implementation uses PyTorch and our experiments are conducted using two Tesla A100 GPUs. The encoder structure consists of two layers, sized 512 and 256, respectively. In contrast, the decoder possesses a structure that mirrors the encoder. The bottleneck layer carries a dimension of 128 while the training mini-batch size is set at 256. Optimization is achieved through Adam, having a learning rate set at 1e-4. Each cell type's default replay number, denoted by m, is set to 20 and a temperature parameter τ is held constant at 1.0. The two loss weight parameters are denoted λ and γ, with both being set at 1.0. During the initial stage, the network is warmed up with L^s_{zinb} over a period of 200 epochs. This is coupled with other classification loss to finetune the network for an additional 200 epochs. Subsequent stages utilize the checkpoint from the terminating stage to initiate the model, and optimization is conducted using suitable losses spanning a total of 200 epochs.

RESULTS COMPARISON

For ease of typesetting, we have incorporated the experimental results of the subsequent benchmarks into the SI.

Intra-data benchmark

Old accuracy comparison

We initiate our analysis by benchmarking scEVOLVE against four distinct baselines using three different classification accuracy metrics across four real datasets: Cao, Quake 10x, Quake Smart-seq2 and Zeisel. The performance of each model, as depicted in Table 1, shows a diminishing trend in accuracy as the stages increase. Notably, individual learning exhibits the most significant performance deterioration, pointing to marked catastrophic forgetting issues. However, scEVOLVE demonstrates minimal impact when incorporating incremental annotation, in comparison to other baselines—with the exception of joint learning. This underscores scEVOLVE's effective algorithmic strategies in combatting forgetting. Notably, the ordinary replay approach, exclusively utilizing cross-entropy loss with exemplars, is insufficient to prevent catastrophic forgetting, resulting in limited old accuracy. Such outcomes can be attributed to the significant discrepancy in exemplar sample size and a potential imbalance in gradient propagation based on current data, leading to catastrophic forgetting. Yet, contrastive replay performs better than ordinary replay, indicating that our proposed prototype-based contrastive learning potentially remedies the imbalance problem. Despite its advantage, contrastive replay still falls short when compared to scEVOLVE, suggesting the significance of uniformly scattered representations and the necessity for the model to generate representations akin to those of joint learning to tackle catastrophic forgetting for optimal performance. As the model learns more cell types, it becomes increasingly challenging to annotate each cell-type sample, resulting in a slight but inevitable decrease in joint learning's old accuracy. Based on our analysis, scEVOLVE ranks second only to joint learning in terms of old accuracy, showcasing its ability to retain previously learned cell types effectively.

scmap, scANVI and CIForm are not originally designed for incremental annotation and only function optimally within fixed cell-type knowledge parameters. Hence, applying these approaches to our benchmark could induce a deterioration in the performance of old accuracy, akin to individual training results. However, a significant improvement in their performance can be observed, particularly for scmap and CIForm, when the simple replay module is integrated, underscoring the efficacy of the sample replay strategy. Although enabling scANVI to handle the incremental annotation task is achieved by this addition, the performance of scANVI bolstered by the ordinary replay strategy remains underwhelming, highlighting the need for the development of more customized algorithms. A comparative assessment of CIForm with ALL and CIForm with OR testifies to the capacity of our comprehensive strategies to more effectively aid CIForm in overcoming the catastrophic forgetting challenge, thus demonstrating its superior performance. Consequently, our proposed modules can be readily incorporated into single-cell annotation methods predicated on deep softmax classification.

New accuracy comparison

The analytical efficacy of scEVOLVE demonstrates superior functionality in terms of annotating current data, surpassing joint learning, but not as competitive when juxtaposed with the other three methodologies. This outcome is anticipatable owing to each method's distinct balance between retaining knowledge of old cell types and assimilating new cell types. As part of its intentional design strategy, scEVOLVE tolerates marginal new accuracy loss to mitigate the issue of catastrophic forgetting. Notably, the anti-forgetting strategy of scEVOLVE can be applied to most existing softmax-based annotation methods through the integration of relevant loss functions. In contrast, joint learning concedes greater new accuracy to retain the memory of old cell types by repetitively learning previously studied samples. Overloading its learning cohort leads to an ultimate loss of focus on acquiring new cell types, thus significantly impeding its new accuracy performance. With the addition of multiple training stages, joint learning also incurs substantial costs due to numerous from-scratch' training sessions involving complete data sets. With new data constantly streaming in, this method of retraining the network each time is unsustainable. On the other hand, individual learning, while offering relatively satisfactory results in annotating new cell types, fails to address the catastrophic forgetting problem, rendering it impractical for the incremental annotation scenario we propose. Likewise, both ordinary replay and contrastive replay offer competitive performance concerning new accuracy. However, their less-than-optimal old accuracy results make them less desirable choices.

Additionally, incorporating traditional replay tactics or our suggested comprehensive strategies into other existing single-cell annotation methods could potentially diminish their new accuracy on the existing dataset. This is reasonably anticipated, given that the storage and replay of old samples can exacerbate the challenge of learning from new samples. However, the application of incremental annotation mechanisms can still enable these methods to yield reasonably competitive results in terms of new accuracy. This suggests that the adverse effect introduced by old samples remains insignificant.

Overall accuracy comparison

The analysis conducted illustrates the necessity of striking a balance between old and new accuracy assessments while employing overall accuracy as the primary evaluation criterion. Notable findings demonstrate that scEVOLVE consistently yields satisfactory outcomes, particularly in the late stages. Despite a potential minor loss of precision in identifying new cell types, scEVOLVE maintains impressive performance levels across all cell types. This signifies an effective compromise between preserving knowledge about old cell types and acquiring proficiency in discerning new ones. On the one hand, joint learning naturally achieves greater overall accuracy as an upper limit compared to other methodologies. However, the marginal difference between the overall accuracy of scEVOLVE and joint learning exceptionally validates scEVOLVE's efficacy. On the other hand, individual learning's limited capacity in dealing with old cell types makes its overall accuracy less competitive, setting the lower limit. Regular and contrastive replay methods fall short in terms of overall accuracy due to their mediocre performance on old cell type's accuracy. This implies an unfavorable trade-off in these approaches. A critical insight gained from comparing scEVOLVE with contrastive replay was the paramount importance of the cell-type decorrelation process. This strategy enhances the segregation of samples from diverse cell types by ensuring homogeneous scattering of each cell-type representation, thereby contributing to mitigating the issue of memory erosion. Comparisons between scEVOLVE and the overall accuracies of other annotation methods utilizing OR and ALL further indicate subpar outcomes. This suggests inherent differences in the ability of varied baseline models to capture the data structure. Contrastive replay methods and cell-type decorrelation strategies, as employed in CIForm with ALL, manifestly yield more beneficial results than ordinary replay like CIForm with OR. In summary, in terms of intra-data experiments, scEVOLVE outperforms other baselines by considerable margins

Table 1: Comparative analysis of performance among diverse baselines in intra-data incremental annotation benchmarking. The term 'OR' denotes Ordinary Replay, while 'ALL' is indicative of Prototypical Contrastive Replay and Cell-Type Decorrelation

	Stage 1			Stage 2			Stage 3			Stage 4		
Quake Smart-seq2	**Old**	**New**	**Overall**	**Old**	**New**	**Overall**	**Old**	**New**	**Overall**	**Old**	**New**	**Overall**
Individual training	–	98.3	98.3	35.6	**98.4**	61.5	0.0	95.2	35.4	7.5	86.6	21.8
Ordinary replay	–	98.3	98.3	73.5	98.0	83.6	79.5	95.0	85.3	80.8	92.0	82.8
Contrastive replay	–	**98.4**	**98.4**	84.1	97.8	89.7	86.6	95.2	89.8	82.7	89.6	83.9
scEVOLVE	–	98.3	98.3	91.0	94.6	92.5	88.2	95.2	90.8	85.5	84.6	85.3
scmap with OR	–	96.9	96.9	85.9	86.1	86.0	83.8	92.6	87.1	85.5	74.8	83.6
scANVI with OR	–	97.2	97.2	41.8	94.0	63.4	23.6	92.6	49.3	19.9	83.6	31.4
CIForm with OR	–	**98.4**	**98.4**	86.2	97.8	91.0	86.0	96.3	89.8	80.5	**93.0**	82.7
CIForm with ALL	–	98.2	98.2	89.1	96.9	92.3	88.8	**96.6**	91.7	85.0	88.5	85.7
Joint learning	–	98.3	98.3	**94.5**	92.4	**93.7**	**92.4**	93.2	**92.7**	**92.0**	77.6	**89.4**
Zeisel	**Old**	**New**	**Overall**	**Old**	**New**	**Overall**	**Old**	**New**	**Overall**	**Old**	**New**	**Overall**
Individual training	–	**99.5**	**99.5**	0.0	**99.9**	45.9	0.0	99.7	44.0	0.0	98.4	10.4
Ordinary replay	–	**99.5**	**99.5**	92.6	**99.9**	96.0	75.3	99.6	86.1	85.0	98.2	86.4
Contrastive replay	–	**99.5**	**99.5**	96.9	**99.9**	98.3	89.4	99.8	94.0	90.9	97.9	91.6
scEVOLVE	–	**99.5**	**99.5**	97.6	**99.9**	98.7	97.2	99.8	98.3	96.8	97.2	96.8
scmap with OR	–	91.6	91.6	85.7	97.9	91.3	89.0	93.7	91.1	89.2	94.7	89.7
scANVI with OR	–	**99.5**	**99.5**	67.0	**99.9**	82.1	65.8	**99.9**	80.8	59.6	98.6	63.7
CIForm with OR	–	99.4	99.4	96.3	**99.9**	98.0	92.7	**99.9**	95.9	87.6	**98.7**	88.8
CIForm with ALL	–	99.4	99.4	97.8	**99.9**	**98.8**	94.0	**99.9**	96.6	89.4	98.1	90.4
Joint learning	–	**99.5**	**99.5**	**99.3**	99.7	99.5	**99.4**	99.2	**99.3**	**99.2**	94.2	**98.7**

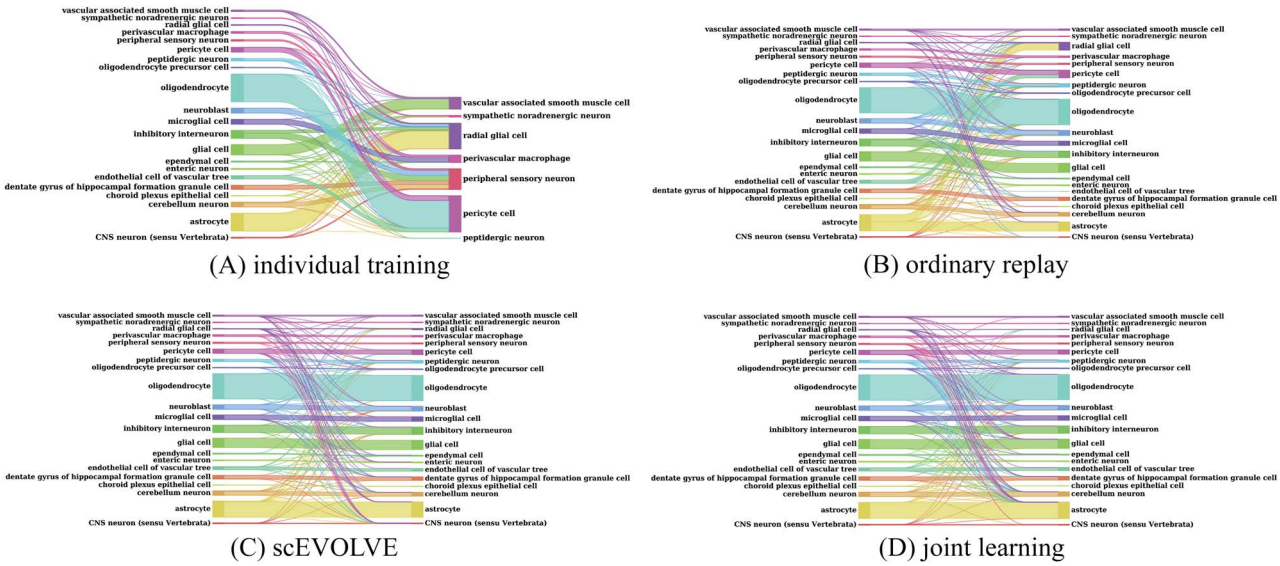

Figure 2. (**A**–**D**) The Sankey mapping plots of test data in the fourth stage for different baselines on the Zeisel dataset. In each plot, the left column represents the true cell type, and the right column represents the predicted cell type. Specific cell-type information can also be seen in the SI.

and effectively circumvents the severe issue of catastrophic forgetting during the incremental annotation of cell types.

Visualization comparison

To unequivocally illustrate the differentiation prediction of various cell types, we have extensively utilized Sankey plots. These plots provide a visual representation of annotation results at the fourth stage of scEVOLVE, as well as other baselines on the Zeisel dataset, as depicted in Figure 2. Our proprietary scEVOLVE system accurately identifies correspondences for both previously known and newly discovered cell types. This substantiates our assertion that our devised tactics effectively impede catastrophic forgetting, while concurrently maintaining superior performance capacity in learning novel cell types. We can further infer that scEVOLVE's overall annotation performance is on par with joint learning, a benchmark for memory retention problems. This is yet another testament to scEVOLVE's significant advancement in addressing incremental annotation challenges. Despite this, the performances of individual and ordinary replay learning methods remain suboptimal. For instance, individual learning tends to forget numerous established cell types, consequently misassigning them to newly identified cell types. Conversely, ordinary replay inaccurately identifies a certain percentage of actrocyte cells as radial glial cells. This discrepancy likely stems from significant differences in the quantity of exemplars and current samples, ultimately leading to a learning imbalance between old and new cell types. This predisposes the system to categorize cells as new types. The observations further attest to the supremacy of scEVOLVE, underlining its efficacy and reliability in predicting diverse cell types.

To enhance the grasp of scEVOLVE's annotation results pertaining to old and new cell types, we have visualized the low-dimensional portrayals of scEVOLVE utilizing t-distributed stochastic neighbor embedding (t-SNE) [50]. This is demonstrated in Figure 3 which presents the t-SNE plots of the testing data across four distinct stages for scEVOLVE on the Zeisel dataset. Each graph set's left subgraph illustrates the classification results based on the understood truth label, while the right subgraph exhibits the classification outcomes stemming from the predicted label. A noteworthy observation is the striking similarity between the classification diagram derived from the predictive label and those from the true label at every stage. This reveals scEVOLVE's capacity to predict the samples with impressive accuracy. Notably, as the training stage evolves, scEVOLVE's discriminatory ability between diverse cell types, both old and new, becomes increasingly evident. This supports scEVOLVE's capability to achieve a balance between enhancing accuracy for previously established types and preserving accuracy for newer types. Furthermore, defined dividing lines are observable between different cell types in the embedding space, a phenomenon that could be attributed to the proposed cell-type decorrelation, aimed at improving the embedding representations, hence aiding cell-type annotation. Overall, it can be asserted that scEVOLVE exhibits outstanding performance with respect to both old and new accuracies within the intra-data experiments.

Inter-tissue benchmark

Three kinds of accuracy comparison

In the subsequent steps, we conducted a series of tests utilizing actual data sets extracted from a variety of tissue types, namely, the Eye, Intestine, Pancreas, and Stomach. These datasets were chosen from a collection of atlas data. The task presented a formidable challenge due to the necessity of grappling with extensive cross-tissue data that possessed a slight batch effect. This required high performance from the algorithm to efficiently process data including striking variances in sample sizes between distinct cell types and the elimination of the batch effect.

Insights from Table 2 demonstrate that scEVOLVE possessed commendable performance in terms of annotation accuracy concerning old cell types; its results were only marginally surpassed by those that utilized the joint learning approach. Interestingly, scEVOLVE emerged as the best approach in achieving the utmost precision concerning new cell types, outperforming all other baseline methods. One of the reasons for this superior performance was that scEVOLVE was adept at shrinking the requirement of replaying old sample sizes. This grants it a higher degree of flexibility to concentrate on new cell types. Notably, scEVOLVE proved to be more efficient in handling large-scale data as compared

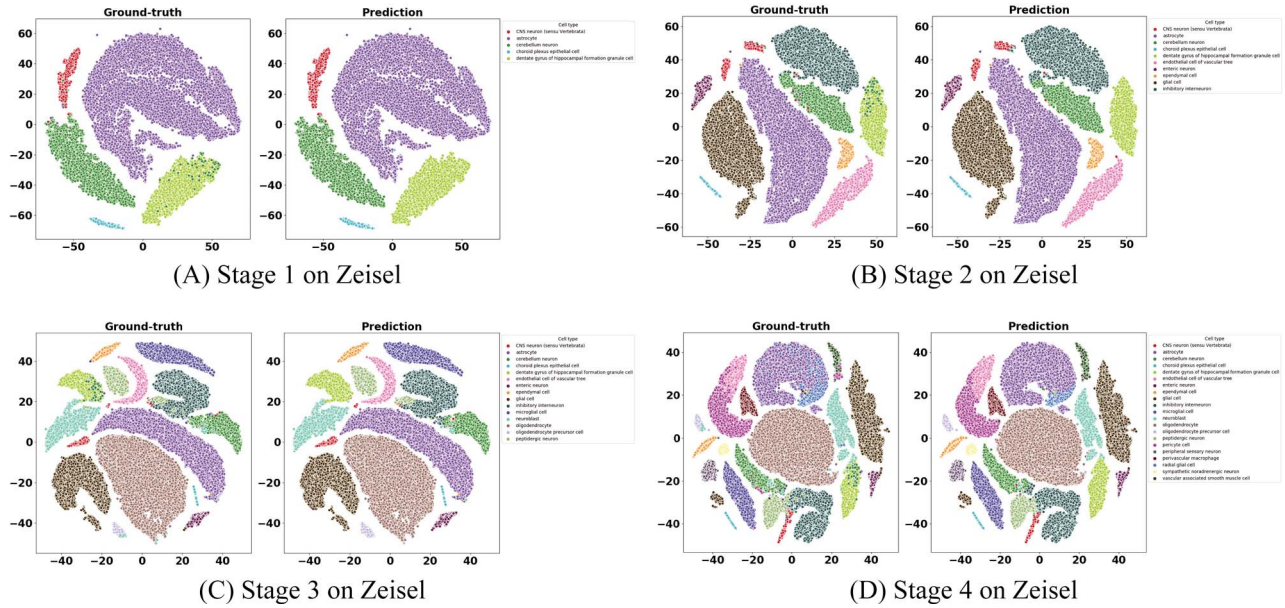

Figure 3. (**A**–**D**) The t-SNE visualization plots of test data in four stages for scEVOLVE on the Zeisel dataset. In each plot, the legend represents a set of cell types, and different colors represent different cell types. Specific cell-type information can also be seen in the SI.

to smaller-scale data. The accuracy of other methodologies, in comparison, dropped significantly. This can be attributed to its prototype-based contrastive learning, which adeptly balances the gradient propagation of both new and old cell types. This helps to address the problem of cell-type imbalance effectively. It further regularizes embedding representations to scatter uniformly which can separate different cell types completely. On the other hand, joint learning is impeded by the sheer volume of data it has to deal with. This directly hinders its learning capacity concerning new cell types, hence leading to an inadequate level of new accuracy. Furthermore, joint learning is also taxing in terms of memory requirements for handling large-scale data. This places considerable pressure on hardware configurations, hence reducing its competitiveness in practical applications. While comparing these performances with scEVOLVE, we noted that the removal of prototypical contrastive learning or cell-type decorrelation would negatively impact overall performance. In essence, this emphasized the importance of these two proposed strategies. In conclusion, given considerations such as remembering and learning new or old cell types, managing large-scale, tissue-based sequential data, and varying learning orders, scEVOLVE showcased robust and superior performance. This underlines its potency in effectively handling inter-tissue incremental annotation benchmark tests.

Inter-data benchmark

Three kinds of accuracy comparison

In this section, we delve deeper into the performance of scEVOLVE and four other benchmark algorithms across four major real datasets in the inter-data investigations. These real datasets are derived from various tissue types, denoted as He, Madisoon, Stewart and Vento. They emanate from diverse atlas data, utilizing different sequencing technologies, and consequently, engender a somewhat augmented batch effect. Analyzing these datasets poses a considerable challenge due to the intricate batch effect that heightens the performance demands of the algorithms. Table 3 presents a level of complexity reflected in the performance of all methods as a result of the batch effect. Notwithstanding, scEVOLVE upholds its high-quality performance, signifying its potential to coherently map cells from contrasting datasets within a single latent space. Furthermore, judging by the old accuracy, its faculty to prevent learning loss remains unaffected by the batch effect, illustrating its robustness. It's plausible that prototype-based contrastive learning could assist in attenuating distribution bias between datasets simplicity by aligning each sample to its own prototypes. On the downside, joint learning has evidently lost some of its competitive edge, particularly in new accuracy. This is due, in part, to its inability to mitigate the batch effect, in addition to having to be retrained on increasingly sizeable datasets, which drastically compromises its adaptability in learning new cell types. Individual learning, because of its unique learning characteristics, is minimally impacted by the batch effect, but its performance leaves much to be desired, especially regarding old accuracy, given its lack of capacity to avoid learning loss. In comparison to scEVOLVE, the old and new accuracy levels of ordinary replay and contrastive replay are inferior, due to the integration of prototypical contrastive learning and cell-type decorrelation. It's also worth noting that scmap's performance in combination with OR produces the least favorable results, implying that the ability to handle batch effects greatly influences the performance in the inter-data benchmark. Despite CIForm with ALL exhibiting superior performance in relation to other methods combined with OR, it still falls short of scEVOLVE's level, suggesting that merely integrating our framework with other annotation methods may not suffice. Fine-tuning specific parameters might be necessary for various basic models. Therefore, scEVOLVE distinguishes itself from other benchmark algorithms through its capacity to effectively counteract the batch effect, making it an ideal and highly competitive solution in real-world scenarios.

Visualization comparison

To more clearly demonstrate scEVOLVE's annotation results and feature representations, a t-SNE visualization plot is presented in Figure 4 following the fourth stage. The initial and subsequent rows in the same figure depict the annotative outcomes from individual learning and scEVOLVE respectively. Significantly, the

Table 2: Comparative analysis of performance among diverse baselines in inter-tissue incremental annotation benchmarking. The term 'OR' denotes Ordinary Replay, while 'ALL' is indicative of Prototypical Contrastive Replay and Cell-Type Decorrelation

	Stage 1 (Intestine)			Stage 2 (Pancreas)			Stage 3 (Stomach)			Stage 4 (Eye)		
Cao_2020	Old	New	Overall	Old	New	Overall	Old	New	Overall	Old	New	Overall
Individual training	–	97.9	97.9	19.6	97.8	56.2	40.0	96.5	44.6	31.5	98.5	48.5
Ordinary replay	–	98.0	98.0	72.7	97.8	84.4	84.8	94.8	85.7	88.2	98.4	90.8
Contrastive replay	–	98.1	98.1	75.9	97.8	86.2	86.6	94.0	87.2	89.8	98.3	91.5
scEVOLVE	–	97.9	97.9	89.0	97.9	93.0	90.1	96.9	90.5	90.4	98.8	92.6
scmap with OR	–	63.9	63.9	53.8	79.8	66.0	58.0	67.9	58.8	60.7	72.9	63.8
scANVI with OR	–	98.5	98.5	20.4	98.2	56.8	73.0	98.2	75.1	70.8	98.6	77.9
CIForm with OR	–	**98.8**	**98.8**	56.4	**98.6**	76.2	80.3	98.0	81.8	80.2	**99.0**	85.0
CIForm with ALL	–	98.4	98.4	66.0	**98.6**	81.3	81.3	**98.3**	82.7	84.6	**99.0**	88.3
Joint learning	–	96.4	96.4	**96.0**	95.2	**95.6**	**95.4**	85.8	**94.6**	**94.4**	96.4	**94.9**
	Stage 1 (Pancreas)			**Stage 2 (Stomach)**			**Stage 3 (Eye)**			**Stage 4 (Intestine)**		
Cao_2020	Old	**New**	**Overall**	**Old**	**New**	**Overall**	**Old**	**New**	**Overall**	**Old**	**New**	**Overall**
Individual training	–	97.7	97.7	35.5	97.2	45.4	14.2	98.6	47.9	21.3	97.9	49.2
Ordinary replay	–	97.7	97.7	90.2	94.7	90.9	88.6	98.7	92.3	84.6	97.6	89.3
Contrastive replay	–	97.9	97.9	93.0	92.9	93.0	90.5	**98.8**	93.8	88.6	97.8	91.9
scEVOLVE	–	98.0	98.0	97.1	97.1	**97.1**	93.3	**98.8**	95.5	90.6	98.3	93.1
scmap with OR	–	84.8	84.8	81.6	72.1	80.1	78.9	73.1	76.6	65.4	57.1	62.3
scANVI with OR	–	97.5	97.5	44.8	96.2	53.0	72.5	98.6	82.9	60.7	**98.4**	74.4
CIForm with OR	–	98.4	98.4	81.1	**98.0**	83.8	75.3	98.7	84.6	68.7	98.1	79.4
CIForm with ALL	–	**98.5**	**98.5**	84.0	97.7	86.2	77.1	98.7	85.7	70.9	98.2	80.8
Joint learning	–	97.3	97.3	**97.2**	85.2	95.3	**95.2**	97.1	**96.0**	**95.5**	95.1	**95.4**

Table 3: Comparative analysis of performance among diverse baselines in inter-data incremental annotation benchmarking. The term 'OR' denotes Ordinary Replay, while 'ALL' is indicative of Prototypical Contrastive Replay and Cell-Type Decorrelation

	Stage 1 (He)			Stage 2 (Madissoon)			Stage 3 (Stewart)			Stage 4 (Vento)		
	Old	New	Overall	Old	New	Overall	Old	New	Overall	Old	New	Overall
Individual training	–	78.7	78.7	2.0	90.9	72.7	37.1	95.1	51.2	22.6	97.9	52.7
Ordinary replay	–	79.2	79.2	78.9	90.9	88.5	83.4	95.4	86.3	72.1	97.6	82.3
Contrastive replay	–	79.5	79.5	79.8	90.9	88.7	85.4	95.1	87.7	75.0	97.2	83.8
scEVOLVE	–	81.7	81.7	**80.9**	**91.2**	**89.1**	86.8	95.7	88.9	79.9	98.2	86.8
scmap with OR	–	75.2	75.2	62.6	83.6	79.3	64.3	91.0	70.8	59.7	89.2	71.5
scANVI with OR	–	80.0	80.0	24.0	88.9	75.6	57.7	93.7	66.4	41.7	96.8	63.7
CIForm with OR	–	85.2	85.2	68.7	**91.2**	86.6	80.6	96.9	84.6	73.2	98.2	83.2
CIForm with ALL	–	**85.4**	**85.4**	70.9	91.0	87.9	81.7	**97.0**	85.2	77.1	**98.3**	85.6
Joint learning	–	78.7	78.7	79.0	90.9	88.5	**88.5**	93.3	**89.7**	**88.6**	96.1	**91.6**
	Stage 1 (Madissoon)			**Stage 2 (Stewart)**			**Stage 3 (Vento)**			**Stage 4 (He)**		
	Old	New	Overall	Old	New	Overall	Old	New	Overall	Old	New	Overall
Individual training	–	90.9	90.9	46.7	95.0	60.6	23.3	98.0	56.2	15.4	78.7	21.3
Ordinary replay	–	91.0	91.0	84.1	95.5	87.4	70.2	97.8	82.3	85.0	79.6	84.5
Contrastive replay	–	91.0	91.0	86.0	95.5	88.7	72.8	97.7	83.8	85.4	80.0	84.9
scEVOLVE	–	**91.3**	**91.3**	87.4	96.2	89.9	79.3	98.0	87.3	88.1	81.6	87.3
scmap with OR	–	85.1	85.1	60.2	93.0	69.6	60.7	90.3	73.7	67.6	75.8	68.3
scANVI with OR	–	89.1	89.1	61.1	90.1	69.4	37.2	96.8	63.5	65.7	82.6	67.3
CIForm with OR	–	91.1	91.1	81.8	**97.2**	86.2	73.6	**98.2**	84.4	87.2	83.5	86.8
CIForm with ALL	–	91.2	91.2	82.5	96.9	86.6	79.4	**98.2**	87.7	88.0	**84.0**	87.7
Joint learning	–	90.9	90.9	**90.8**	93.0	**91.4**	**90.7**	95.8	**92.9**	**92.9**	79.2	**91.6**

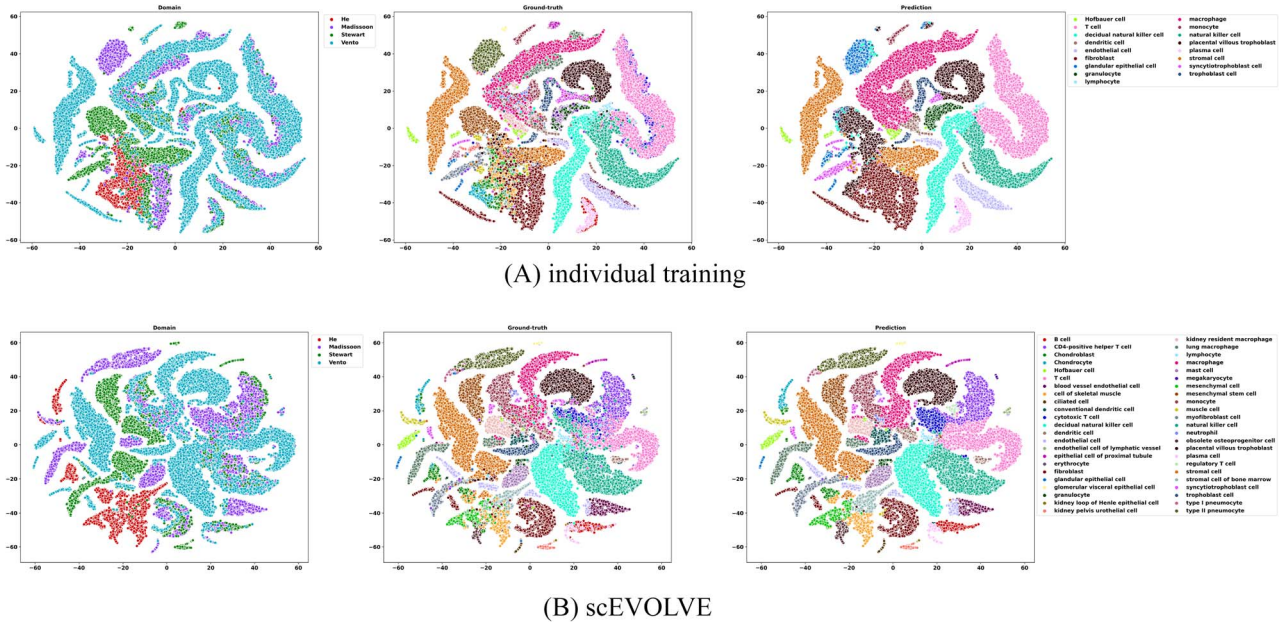

Figure 4. (**A**, **B**) The t-SNE visualization plots of test data in four stages for individual training and scEVOLVE on sequential He, Madissoon, Stewart, and Vento datasets. In each plot, the legend of the left subgraph represents the dataset set, and the middle subgraph and the right subgraph share the same legend, representing the cell-type set. Specific cell-type information can also be seen in the SI.

leftmost image, demonstrating the division of samples across various domains, presumably indicates the varying capabilities of different methods to eliminate batch effects. A comparison reveals scEVOLVE's superior ability in distinguishing data from different domains in contrast to individual learning, which tends to amalgamate them, particularly evident in the case of Madissoon and Vento. This attribute of scEVOLVE may stem from its capacity to harmonize samples with their prototypes and to enhance the embedding of representations. These attributes further assist the algorithm in learning a more effective latent space and effectively eliminating the batch effect. The central and right-sided images represent the two-dimensional visualization plots drawn from the ground-truth labels and prediction labels correspondingly. scEVOLVE demonstrates a clear capability to differentiate between varying cell types. This primarily owes to its regularization of embedding representations to scatter uniformly at every stage and its ability to maintain a balance in the gradient propagation. These features help the model in recalling and annotating samples of previous cell types. Conversely, the latent space evolved from individual learning fails to cultivate a proper and unique cluster structure, thereby vividly demonstrating catastrophic forgetting. Thus, it can be concluded that scEVOLVE effectively addresses this problem of forgetting and provides superior performance overall.

Robustness analysis

Accuracy variation in inter-tissue benchmark

In order to evaluate the resilience of various procedures when confronting the variability of data learned at different stages, we reconfigured the sequence of large-scale data learned in the four stages and assessed the performance of scEVOLVE and additional benchmarks. The corresponding experimental results from datasets within the inter-tissue benchmark are presented in Figures 5 and 6. Notably, individual learning displays significantly lower performance on old accuracy compared to other methods, thus, to make the comparative performance evaluation of other methods more distinct, it was excluded from Figure 5. For clarity in presentation, the following terms are used in the legends of the figures: individual, 'ordinary', 'contrastive' and 'joint' to represent individual learning, ordinary replay, contrastive replay and joint learning, respectively. It is observed that the overall performance of all methods tends to be less stable regarding new accuracy, in comparison to old accuracy. This can be primarily attributed to the fact that old cell types were learned before the finalization of the learned embedding space and the corresponding parameters that might directly influence the learning process of the new cell type. As depicted in Figures 5 and 6, the performance of scEVOLVE remains relatively stable for both old and new accuracy, irrespective of the learning sequence variations of datasets, demonstrating its robustness. scEVOLVE's embedding representation mimics that of joint learning, hence, it remains rather insulated from the learning sequence of the data. Additionally, its strategy to prevent memory fading by managing gradient propagation aims to balance the learning of both old and new cell types, thereby promising equal treatment by the model for both. In contrast, violent fluctuations are observed in the old accuracy of individual learning and the new accuracy of joint learning, underscoring their instability. The performance of ordinary replay also has shortcomings, indicating that merely reducing memory fading by storing exemplars is a weak strategy that leaves its accuracy easily susceptible to changes in the learning order of the dataset. Similarly, contrastive replay often provides sub-optimal results, being too dependent on the learning order of datasets due to a lack of stability control in representation learning; the embedding features of the current stage are altered along with the variation of features from the previous stage. In summary, scEVOLVE consistently showcases robust performance notwithstanding different sequencing of datasets in the inter-tissue benchmarks.

Accuracy variation in inter-data benchmark

In an effort to examine the model's resilience to alterations in the learning sequence of datasets under the batch effect, we evaluated the annotation accuracy fluctuations of scEVOLVE in comparison to other methods using the inter-data benchmark.

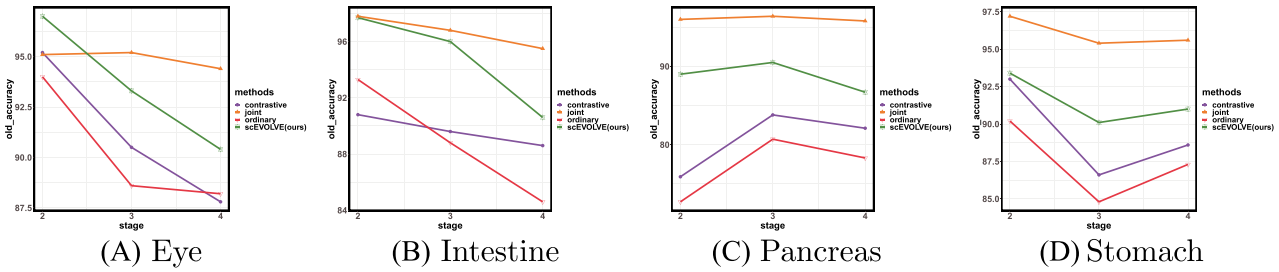

Figure 5. Line graph of the variation in old accuracy of the four methods when the four datasets from different tissues are learned at different stages. In each graph, the legend represents the set of comparison methods, namely 'contrastive', 'joint', 'ordinary' and 'scEVOLVE (ours)'.

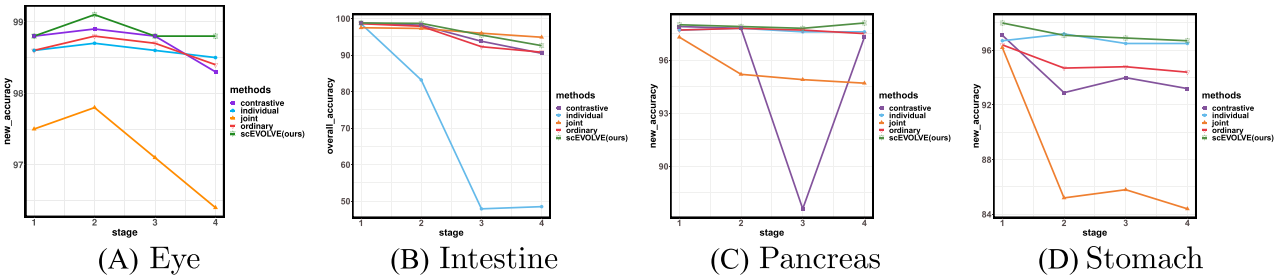

Figure 6. Line graph of the variation in new accuracy of the five methods when the four datasets from different tissues are learned at different stages. In each graph, the legend represents the set of comparison methods, namely 'contrastive', 'individual', 'joint', 'ordinary' and 'scEVOLVE (ours)'.

Figures 7 and 8 demonstrate the trajectories of both old and new accuracy across all analyzed methods. We chose not to graph the old accuracy line for individual learning due to its inferior performance. Primarily, we observed that the old and new accuracy of all methods exhibited greater instability compared to those displayed in the inter-tissue benchmark. This is to be expected, given that the pronounced batch effect presents a more significant obstacle to the incremental learning of models. In parallel to occurrences witnessed with minor batch effect incidents, scEVOLVE consistently yielded stable findings, as displayed in Figures 7 and 8. Subsequently, despite the presence of notable batch effects, these findings reveal that scEVOLVE preserves satisfactory and robust old and new accuracy across the compared methods, underscoring its effectiveness and resilience in counteracting batch effects. In contrast, joint learning's new accuracy performance was the weakest amongst all the methods, showcasing violent fluctuations. This could be ascribed to the necessity of retraining all samples to acquire new cell types at every stage, leading to heightened sensitivity to the learning order of the dataset. Nonetheless, while ordinary replay and contrastive replay demonstrated stability akin to scEVOLVE, their performance was inferior when matched against scEVOLVE's aptitude for controlling gradient propagation and standardizing embedding representation, both of which are paramount for reminiscing old cell types and assimilating new ones. In summary, when juxtaposed with other baselines, scEVOLVE can effectively cushion the impact of batch effects. Furthermore, varying the learning sequence of the dataset has minimal influence on its performance regarding both old and new accuracy.

Discussion of hyperparameters

Sensitivity analysis for exemplar size

In this section, we explore the impact of individual hyperparameter values on the performance of the model. We initially turn our attention to m, a hyperparameter that dictates the quantity of exemplars retained for each cell type. This hyperparameter plays a crucial role in the performance enhancement of scEVOLVE. Increasing m has a dual effect. On the positive side, a higher value of m lets the exemplars encapsulate a richer array of data pertaining to established cell types, thereby elevating the model's accuracy in relation to these old cell types. Conversely, a higher m value also imposes a greater computational strain on the model, increasing both the required learning capacity and computational resources. This necessitates the model to allocate additional time and memory for mastering larger exemplar sets, conversely depriving it of the time to comprehend new cell types. This inevitably leads to a drop in accuracy when encountering new cell types. Consequently, choosing the optimal m value requires a careful balance between retaining knowledge about established cell types and accommodating the learning of new cell types. This underscores the necessity of analyzing the impact of m on the model's precision.

In our study, we directed investigations on two distinct datasets: Quake 10x and Quake Smart-seq2. The changing patterns of the overall precision of scEVOLVE at the fourth stage are visually represented in both Figures 9(A) and 9(B). To focus on instances where the increase in m oscillates between the spectrum of $[10, 15, 20, 25, 30]$, we observed a consistent rise in the total precision of each method as the value of m escalated. This infers that assessing and calibrating the model's accuracy against its computational load must be carried out with practical applications in mind. Particularly when m manifests a smaller value, for example, 10, scEVOLVE is competent in delivering superior performance. This reinforces its dominance in maintaining equilibrium between the education of established and emerging cell types. While ordinary replay and contrastive replay's executions uplift with the growth of m, the outcome is not as optimal as scEVOLVE. As a general rule, we regularly established m at 20. This approach has proven effective in ensuring a balance between precision and computational load, and it has reinforced the superior performance of the scEVOLVE method in our experiments.

Sensitivity analysis for labeled ratio

The hyperparameter known as the labeled ratio in the model governs the proportion of data designated for training and testing.

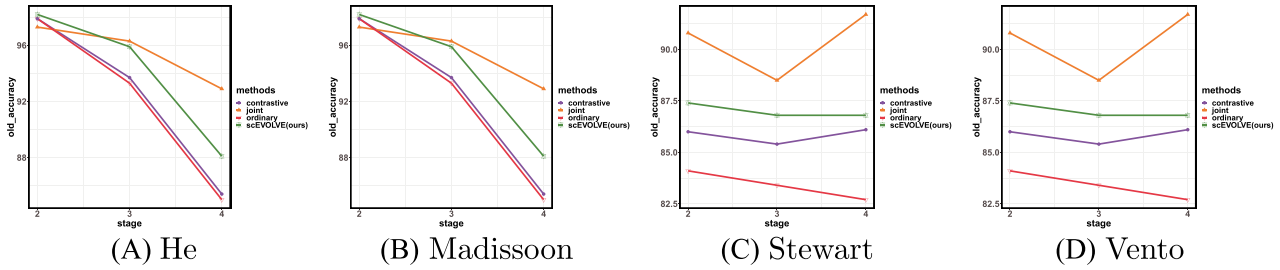

Figure 7. Line graph of the variation in old accuracy of the four methods when the four datasets with batch effects are learned at different stages. In each graph, the legend represents the set of comparison methods, namely 'contrastive', 'joint', ordinary' and 'scEVOLVE (ours)'.

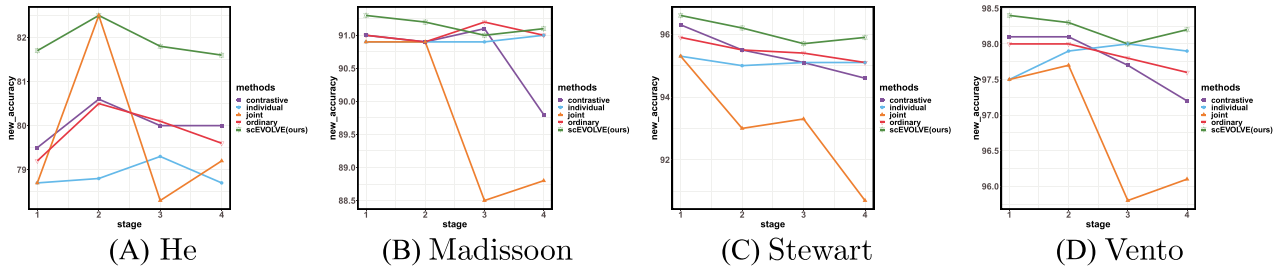

Figure 8. Line graph of the variation in new accuracy of the five methods when the four datasets with batch effects are learned at different stages. In each graph, the legend represents the set of comparison methods, namely 'contrastive', individual', 'joint', ordinary' and 'scEVOLVE (ours)'.

Essentially, a greater labeled ratio equates to a larger quantity of data employed as training data within the dataset. From this, we extrapolate that an elevated labeled ratio could potentially enhance new accuracy as it facilitates deeper learning of the existing training data relevant to the current cell type. However, this scenario might inversely affect the old accuracy. An increase in training data may intensify the inequity between the model's comprehension of new and old cell types, thus impeding its capability to recall old cell types. Consequently, it becomes imperative to investigate the effects of varying labeled ratios on the model's precision to optimize its accuracy.

Figure 9(C) and (D) distinctly illustrates the shifting trends of overall precision during the fourth phase of scEVOLVE and other baseline parameters when the labeled ratio alters on Quake 10x and Quake Smart-seq2 datasets, respectively. Our analysis synthesizes instances where the incremental change of the labeled ratio spans within the scope of $[0.05, 0.075, 0.1, 0.125, 0.2]$. Observations suggest that the integral accuracy of scEVOLVE remains relatively unwavering in relation to the labeled ratio, underscoring that our model is only minimally influenced by the magnitude of the labeled ratio. Importantly, scEVOLVE consistently exhibits commendable performance compared to other studied methods, thereby validating its preeminence in curtailing the oblivion of antique cell types while facilely learning novel cell types. A slight elevation in the comprehensive accuracy is noticeable in the context of joint learning as the labeled ratio surges, signifying its growing need for an expanded body of training data. Contrastingly, the performance trajectories of both ordinary and contrastive replay display a clear decrement as the labeled ratio heightens. This suggests that an enlarged set of training data may augment the difficulty of equilibrating the learning process between conventional and contemporary cell types. For the purposes of this paper, we have defaulted the labeled ratio to 0.1.

Sensitivity analysis for stage number

It has been established that multiple learning stages in the model could potentially increase the propensity to forget old cell types. Similarly, integrating excessive cell types for learning purposes might compromise the model's capacity to acquire new cell types. Consequently, it is imperative to examine the performance of scEVOLVE and other fundamental benchmarks in relation to an expanded number of stages. Empirical experiments have been performed on two large-scale, real data sets, Quake 10x and Quake Smart-seq2, featuring nine learning stages. Figure 10(A) and (B), respectively, illustrates the alterations in total accuracy across five methodologies related to these data sets. However, as individual learning significantly lags due to its inherent inability to prevent cell-type forgetting, these trends have not been graphically represented for streamlined observation. Specifically, under these stringent conditions, a noticeable downward shift in overall accuracy across all methodologies was observed, aligning with our assumption of increased difficulty in preventing forgetting old cell types while assimilating new ones as stage count increases. Amongst various models, joint learning was the most resilient, demonstrating its dominance over incremental learning associated with a higher number of stages, despite its declining competitiveness due to memory demand saturation. The moderation in variation trend displayed by scEVOLVE, although secondary to joint learning, further corroborates its efficacy in averting catastrophic forgetting and assimilating new cell types, even amidst escalating stages of incremental learning. Comparatively, the decrement trend of both ordinary and contrastive replay is quite abrupt, rendering them less efficient than scEVOLVE considering its superior capacity to regulate gradient propagation and standardize embedding representation. To summarize, scEVOLVE can efficiently withstand the load of extra stages, thus demonstrating its eligibility and promising performance, making this model a pragmatic choice for real-world situations.

Sensitivity analysis for temperature and loss weight

To demonstrate the extent of sensitivity that scEVOLVE possesses towards temperature variations denoted as τ, we executed a series of control experiments using Cao, Quake 10x and Quake

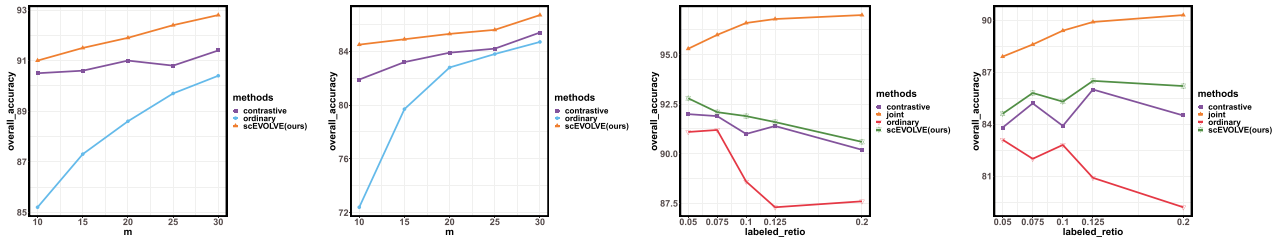

Figure 9. Overall accuracy on Quake 10x and Quake Smart-seq2 datasets. (**A**, **B**) Changing the number of exemplars for each cell type, and the legend represents the set of comparison methods, namely 'contrastive', 'ordinary' and 'scEVOLVE (ours)'; (**C**, **D**) changing the labeled ratio, and the legend represents the set of comparison methods, namely 'contrastive', 'joint', 'ordinary' and 'scEVOLVE (ours)'.

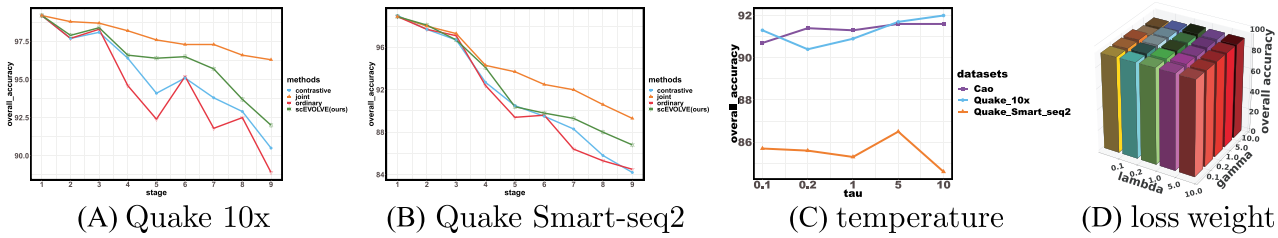

Figure 10. (**A**, **B**) The variations of overall accuracy in nine stages on Quake 10x and Quake Smart-seq2 datasets, and the legend represents the set of comparison methods, namely 'contrastive', 'joint', 'ordinary' and 'scEVOLVE (ours)'; (**C**, **D**) the overall accuracy variations of scEVOLVE with the changes of temperature τ, weight λ and γ on the Cao, Quake 10x and Quake Smart-seq2 datasets in the last (fourth) stage. The legend in (c) represents the set of tested datasets.

Table 4: The recommended usage range of all hyperparameters in our model. 'ratio' and 'stage' refer to labeled ratio and stage number, respectively

	m	Ratio	Stage	τ	λ	γ
Choice	[20, 30]	[0.075, 0.125]	[2, 6]	[0.2, 5.0]	[0.2. 5.0]	[0.2, 5.0]

Smart-seq2 datasets. The outcomes of these experiments have been visually represented in Figure 10(C). Throughout a substantial range of τ values, specifically $\tau \in [0.1, 0.2, 1.0, 5.0, 10.0]$, there was no significant alteration in performance exceeding 1%. This evidence suggests the robustness of scEVOLVE in relation to temperature variations denoted by τ. Moreover, an analysis of scEVOLVE's performance response to modifications in the loss weight parameters λ and γ was carried out on the Cao dataset. The results depicted in Figure Figure 10(D) reveal that the overall accuracy remains relatively consistent amidst changes in $\lambda \in [0.1, 0.2, 1.0, 5.0, 10.0]$ and $\gamma \in [0.1, 0.2, 1.0, 5.0, 10.0]$. This affirms the stability of scEVOLVE with respect to adjustments made for the parameters λ and γ.

Finally, in order to facilitate other researchers to use our method, we recommend some appropriate hyperparameter choice ranges in Table 4. In addition, we still suggest that the selection of hyperparameters should depend on specific datasets and experimental resources.

CONCLUSION

This article introduces a novel, pragmatic and demanding task of cell-type incremental annotation in the domain of single-cell research, a task that takes into account real-world cell annotation needs. This task presents a unique challenge of only having access to data from the current stage, thereby rendering the data from prior stages unavailable. This unavailability might potentially lead to severe forgetting issues in the model. To remediate this pressing concern, the scEVOLVE, an innovative continual learning framework that excludes forgetting, is tailored. This framework is a trifold design. Firstly, a memory buffer is structured to conserve select old samples employing the nearest prototype exemplar selection mechanism, thus curtailing catastrophic forgetting. Secondly, considering the prevalent cell-type imbalance issue, the framework proposes a novel prototype-based contrastive replay loss as an alternative to the traditional cross-entropy to achieve balanced gradient propagation. Lastly, to elevate model performance from a representational perspective, the framework introduces cell-type decorrelation regularization. The motive behind this inclusion is to stimulate representations to be distributed more evenly in the associative embedding space, mirroring the principles of joint learning. It is paramount to note that this incremental learning framework is entirely independent yet complementary to the foundational concepts of most deep softmax-based cell annotation algorithms. By integrating this framework with existing algorithms, one can attain superior outcomes.

To thoroughly assess the performance of the algorithm, meticulous construction of comprehensive benchmarks and baselines was done. Explicitly, scEVOLVE was put up against four competitive methods—individual learning, ordinary replay, contrastive replay and joint learning, within three rigorous scenarios namely intra-data, inter-tissue, and inter-data. Firstly, the intra-data trials revealed the proficiency of scEVOLVE, ranking second to joint learning in terms of old accuracy, but outperforming all

others in new accuracy. This underscored the effectiveness of its strategies against forgetting. Furthermore, scEVOLVE epitomized an optimal equilibrium between preventing knowledge forgetting and learning new cell types. Meanwhile, drawing from the observation that joint learning tends to yield sub-optimal results when it comes to new accuracy, it can be inferred that joint learning is not always the optimal choice, even when memory capacity is ample. Rather, incremental learning like our strategies may produce superior outcomes. This assertion is validated by instances where scEVOLVE has achieved parity in accuracy with joint learning. Furthermore, the notable enhancement in accuracy when our framework is integrated with existing algorithms, such as scmap, scANVI and CIForm, amplifies the value of our framework. This notable boost in performance encourages the expansion of our framework's application to a broader range of existing models, thereby enhancing their applicability in practical scenarios. Secondly, in the inter-tissue category, scEVOLVE exhibited increased competitiveness with mass-scale data featuring subtle batch effects, surpassing its previous small-scale data performance. Simultaneously, the efficiency of all other methods witnessed a steep and inevitable decline, consolidating scEVOLVE's potential to maneuver mass-scale data in real-world situations. It is worth highlighting that the overall accuracy of scEVOLVE surpasses that of joint learning in certain datasets. This underscores the efficacy of our approach, namely prototypical contrastive replay learning, in mitigating the batch effect. Thirdly, our inter-data trials were designed to evaluate the performance of scEVOLVE, alongside other baselines, on mass-scale data enveloped with severe batch effects. The results further distinguished scEVOLVE from other methods by a noticeable margin, thereby underlining its supremacy. Besides, the orderly arrangement of a substantial variety of cell types on a two-dimensional plane further demonstrates that our cell-type decorrelation learning strategy can enhance a more uniform representation of these cell types. Fourthly, scEVOLVE's robustness and choice of hyperparameter values were examined in added experiments. The results indicate that scEVOLVE exhibits superior stability due to its ability to effectively manage data diversity at various learning stages and hyperparameter selections. This characteristic enhances its reproducibility and promotes its broader application. We have empirically validated that scEVOLVE maintains competitive performance when the training stage duration is extended from 4 to 9, further substantiating its capacity for incremental learning across a multitude of cell types over an extended timeframe. Additionally, we successfully expanded the scope of tissues studied from 4 to 8 in the SI, revealing that scEVOLVE continues to maintain high accuracy with minimal forgetting, thus preserving its efficacy. To culminate, scEVOLVE showcased an adept capability in solving the cell-type incremental annotation challenge. The implications of our findings could prove beneficial for tangible applications such as updates and upgrades of cell annotation systems.

LIMITATIONS AND FUTURE IMPLICATIONS

In this paper, we pioneer the conceptualization and development of single-cell incremental annotation as a solution to the issue of catastrophic forgetting. We examine the problem within the context of an optimal close-set setting, denoted as $\mathcal{C}^{it} \subseteq \mathcal{C}^{ir}$. Nonetheless, the real world presents more complex scenarios such as the open-set and open-partial settings. As a result, our future work will involve enhancing our model to be adaptable to these more demanding situations. Furthermore, our approach efficiently mitigates the problem of forgetting through the storage of exemplars. However, our key aim is to create replay-free algorithms that eliminate the necessity for exemplars entirely.

While the scEVOLVE model is capable of gradually acquiring knowledge about new cell types without severely compromising knowledge of the existing types, it is necessary to address its apparent limitations. Relying solely on scRNA-seq data for cell annotation may prove deficient due to potential gaps in information or noise. As a solution, we suggest broadening the incremental annotation challenge to encompass other omics or cell morphology. This would provide a more nuanced representation of cellular heterogeneity from various viewpoints. In addition, it's worth noting that scRNA-seq data only capture a static snapshot at a specific moment in time, hence it lacks temporal information. To rectify this, we propose tailoring our task and suggested framework for time series data, which would enable us to investigate the dynamic trajectory of cell development in subsequent studies.

Key Points

- We introduce an innovative, feasible and demanding endeavor named cell-type incremental annotation. Additionally, we have developed an avant-garde technique, called scEVOLVE, which is skillfully tailored to execute this task with utmost efficacy.
- Our scEVOLVE algorithm is engineered to preserve specific historical data within a memory buffer, an action designed for subsequent replay throughout future training phases. In an effort to effectively address the issue of cell-type imbalance, scEVOLVE innovatively establishes a prototypical contrastive learning objective.
- To replicate the training of the oracle model alongside comprehensive data, scEVOLVE implements the principle of cell-type decorrelation. This principle notably enhances the uniform distribution of feature representations for each cell type.
- Comprehensive studies conducted on meticulously crafted benchmarks reveal that scEVOLVE efficiently mitigates the issue of catastrophic forgetting. Furthermore, it has demonstrated the capacity to progressively learn numerous cell types over an extended duration.

SUPPLEMENTARY DATA

Supplementary data are available online at https://academic.oup.com/bib.

FUNDING

This work was supported by the National Key Research and Development Program of China (2021YFF1200902) and the National Natural Science Foundation of China (32270689, 12126305).

REFERENCES

1. Kolodziejczyk AA, Kim JK, Svensson V, *et al.* The technology and biology of single-cell RNA sequencing. *Mol Cell* 2015;**58**(4):610–20.
2. Ding J, Adiconis X, Simmons SK, *et al.* Systematic comparison of single-cell and single-nucleus RNA-sequencing methods. *Nat Biotechnol* 2020;**38**(6):737–46.

3. Lähnemann D, Köster J, Szczurek E, *et al.* Eleven grand challenges in single-cell data science. *Genome Biol* 2020;**21**(1):1–35.
4. Yan W, Zhang K. Tools for the analysis of high-dimensional single-cell RNA sequencing data. *Nat Rev Nephrol* 2020;**16**(7): 408–21.
5. Shao X, Liao J, Xiaoyan L, *et al.* scCATCH: automatic annotation on cell types of clusters from single-cell RNA sequencing data. *Iscience* 2020;**23**(3):100882.
6. Bao W, Lin X, Yang B, Chen B. Gene regulatory identification based on the novel hybrid time-delayed method. *Front Genet* 2022;**13**:888786.
7. Bao W, Yujian G, Chen B, Huiping Y. Golgi_df: Golgi proteins classification with deep forest. *Front Neurosci* 2023;**17**: 1197824.
8. Abdelaal T, Michielsen L, Cats D, *et al.* A comparison of automatic cell identification methods for single-cell RNA sequencing data. *Genome Biol* 2019;**20**(1):1–19.
9. Wolf FA, Angerer P, Theis FJ. SCANPY: large-scale single-cell gene expression data analysis. *Genome Biol* 2018;**19**:1–5.
10. Chen L, Zhai Y, He Q, *et al.* Integrating deep supervised, self-supervised and unsupervised learning for single-cell RNA-seq clustering and annotation. *Genes* 2020;**11**(7):792.
11. Kiselev VY, Yiu A, Hemberg M. scmap: projection of single-cell RNA-seq data across data sets. *Nat Methods* 2018;**15**(5): 359–62.
12. Satija R, Farrell JA, Gennert D, *et al.* Spatial reconstruction of single-cell gene expression data. *Nat Biotechnol* 2015;**33**(5): 495–502.
13. Chenling X, Lopez R, Mehlman E, *et al.* Probabilistic harmonization and annotation of single-cell transcriptomics data with deep generative models. *Mol Syst Biol* 2021;**17**(1):e9620.
14. Brbić M, Zitnik M, Wang S, *et al.* Mars: discovering novel cell types across heterogeneous single-cell experiments. *Nat Methods* 2020;**17**(12):1200–6.
15. Lotfollahi M, Naghipourfar M, Luecken MD, *et al.* Mapping single-cell data to reference atlases by transfer learning. *Nat Biotechnol* 2022;**40**(1):121–30.
16. Rebuffi SA, Kolesnikov A, Sperl G, Lampert CH. iCaRL: incremental classifier and representation learning. *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, p. 2001–10, 2017.
17. Li Z, Hoiem D. Learning without forgetting. *IEEE Trans Pattern Anal Mach Intell* 2017;**40**(12):2935–47.
18. Goodfellow IJ, Mirza M, Da Xiao AC, Bengio Y. An empirical investigation of catastrophic forgetting in gradient-based neural networks. arXiv preprint arXiv:1312.6211, 2013.
19. Kirkpatrick J, Pascanu R, Rabinowitz N, *et al.* Overcoming catastrophic forgetting in neural networks. *Proc Natl Acad Sci* 2017;**114**(13):3521–26.
20. Rolnick D, Ahuja A, Schwarz J, *et al.* Experience replay for continual learning. *Adv Neural Inf Process Syst* 2019;**32**:348–358.
21. Lopez-Paz D, Ranzato MA. Gradient episodic memory for continual learning. *Adv Neural Inf Process Syst* 2017;**30**:6467–6476.
22. Rannen A, Aljundi R, Blaschko MB, Tuytelaars T. Encoder based lifelong learning. *Proceedings of the IEEE International Conference on Computer Vision*, p. 1320–1328, 2017.
23. Liu X, Masana M, Herranz L, *et al.* Rotate your networks: better weight consolidation and less catastrophic forgetting. *2018 24th International Conference on Pattern Recognition (ICPR)*, p. 2262–8. IEEE, 2018.
24. Rusu AA, Rabinowitz NC, Desjardins G, *et al.* Progressive neural networks. arXiv preprint arXiv:1606.04671, 2016.
25. Mallya A, Lazebnik S. PackNet: adding multiple tasks to a single network by iterative pruning. *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, p. 7765–73, 2018.
26. De Lange M, Aljundi R, Masana M, *et al.* A continual learning survey: defying forgetting in classification tasks. *IEEE Trans Pattern Anal Mach Intell* 2021;**44**(7):3366–85.
27. Masana M, Liu X, Twardowski B, *et al.* Class-incremental learning: survey and performance evaluation on image classification. *IEEE Trans Pattern Anal Mach Intell* 2022;**45**(5):5513–33.
28. Alquicira-Hernandez J, Sathe A, Ji HP, *et al.* scPred: accurate supervised method for cell-type classification from single-cell RNA-seq data. *Genome Biol* 2019;**20**(1):1–17.
29. Cao Z-J, Wei L, Shen L, *et al.* Searching large-scale scRNA-seq databases via unbiased cell embedding with cell blast. *Nat Commun* 2020;**11**(1):3458.
30. Yang F, Wang W, Wang F, *et al.* scBERT as a large-scale pretrained deep language model for cell type annotation of single-cell RNA-seq data. *Nat Mach Intell* 2022;**4**(10):852–66.
31. Chen L, He Q, Zhai Y, Deng M. Single-cell RNA-seq data semi-supervised clustering and annotation via structural regularized domain adaptation. *Bioinformatics* 2021;**37**(6): 775–84.
32. Bae S, Na KJ, Koh J, *et al.* CellDART: cell type inference by domain adaptation of single-cell and spatial transcriptomic data. *Nucleic Acids Res* 2022;**50**(10):e57–7.
33. Zhai Y, Chen L, Deng M. scGAD: a new task and end-to-end framework for generalized cell type annotation and discovery. *Brief Bioinform* 2023;**24**(2):bbad045.
34. Zhai Y, Chen L, Deng M. Generalized cell type annotation and discovery for single-cell RNA-seq data. *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. **37**, p. 5402–10, 2023.
35. Zhai Y, Chen L, Deng M. Realistic cell type annotation and discovery for single-cell RNA-seq data. *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, p. 4967–74, 2023.
36. Eraslan G, Simon LM, Mircea M, *et al.* Single-cell RNA-seq denoising using a deep count autoencoder. *Nat Commun* 2019;**10**.
37. He K, Fan H, Wu Y, XIe S, Girshick R. Momentum contrast for unsupervised visual representation learning. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, p. 9729–38, 2020.
38. Khosla P, Teterwak P, Wang C, *et al.* Supervised contrastive learning. *Adv Neural Inf Process Syst* 2020;**33**:18661–73.
39. Hua T, Wang W, Xue Z, Ren S, Wang Y, Zhao H. On feature decorrelation in self-supervised learning. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, p. 9598–608, 2021.
40. Huang L, Yang D, Lang B and Deng J. Decorrelated batch normalization. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, p. 791–800, 2018.
41. Cao J, Packer JS, Ramani V, *et al.* Comprehensive single-cell transcriptional profiling of a multicellular organism. *Science* 2017;**357**(6352):661–7.
42. Schaum N, Karkanias J, Neff NF, *et al.* Single-cell transcriptomics of 20 mouse organs creates a Tabula Muris: the Tabula Muris Consortium. *Nature* 2018;**562**(7727):367.
43. Zeisel A, Hochgerner H, Lönnerberg P, *et al.* Molecular architecture of the mouse nervous system. *Cell* 2018;**174**(4):999–1014.e22.
44. Cao J, O'Day DR, Pliner HA, *et al.* A human cell atlas of fetal gene expression. *Science* 2020;**370**:808–850.
45. He J, Yan J, Wang J, *et al.* Dissecting human embryonic skeletal stem cell ontogeny by single-cell

transcriptomic and functional analyses. *Cell Res* 2021;**31**(7): 742–57.
46. Madissoon E, Wilbrey-Clark A, Miragaia RJ, *et al.* scRNA-seq assessment of the human lung, spleen, and esophagus tissue stability after cold preservation. *Genome Biol* 2020;**21**(1):1–16.
47. Stewart BJ, Ferdinand JR, Young MD, *et al.* Spatiotemporal immune zonation of the human kidney. *Science* 2019;**365**(6460): 1461–6.
48. Vento-Tormo R, Efremova M, Botting RA, *et al.* Single-cell reconstruction of the early maternal–fetal interface in humans. *Nature* 2018;**563**(7731):347–53.
49. Jing X, Zhang A, Liu F, *et al.* CIForm as a transformer-based model for cell-type annotation of large-scale single-cell RNA-seq data. *Brief Bioinform* 2023;bbad195.
50. Van der Maaten L, Hinton G. Visualizing data using t-SNE. *J Mach Learn Res* 2008;**9**(11).