Generalized Cell Type Annotation and Discovery for Single-Cell RNA-Seq Data

Yuyao Zhai[1*], **Liang Chen**[4*], **Minghua Deng**[1, 2, 3†]

[1] School of Mathematical Sciences, Peking University
[2] Center for Statistical Science, Peking University
[3] Center for Quantitative Biology, Peking University
[4] Huawei Technologies Co., Ltd.
zhaiyuyao@stu.pku.edu.cn, chenliang260@huawei.com, dengmh@pku.edu.cn

Abstract

The rapid development of single-cell RNA sequencing (scRNA-seq) technology allows us to study gene expression heterogeneity at the cellular level. Cell annotation is the basis for subsequent downstream analysis in single-cell data mining. Existing methods rarely explore the fine-grained semantic knowledge of novel cell types absent from the reference data and usually susceptible to batch effects on the classification of seen cell types. Taking into consideration these limitations, this paper proposes a new and practical task called generalized cell type annotation and discovery for scRNA-seq data. In this task, cells of seen cell types are given class labels, while cells of novel cell types are given cluster labels instead of a unified "unassigned" label. To address this problem, we carefully design a comprehensive evaluation benchmark and propose a novel end-to-end algorithm framework called scGAD. Specifically, scGAD first builds the intrinsic correspondence across the reference and target data by retrieving the geometrically and semantically mutual nearest neighbors as anchor pairs. Then we introduce an anchor-based self-supervised learning module with a connectivity-aware attention mechanism to facilitate model prediction capability on unlabeled target data. To enhance the inter-type separation and intra-type compactness, we further propose a confidential prototypical self-supervised learning module to uncover the consensus category structure of the reference and target data. Extensive results on massive real datasets demonstrate the superiority of scGAD over various state-of-the-art clustering and annotation methods.

Introduction

As more and more well-annotated scRNA-seq reference data become available, many new automatic annotation methods sprung up in order to simplify the cell annotation process on unlabeled target data (Cao et al. 2019; Chen et al. 2020b,c; Yuan, Chen, and Deng 2022). On this basis, assume that $\mathcal{C}_r$ and $\mathcal{C}_t$ represent the label sets of reference data and target data, respectively. In earlier developed cell annotation methods, it is assumed that all cell types in the target data need to be present in the reference data, that is $\mathcal{C}_t \subseteq \mathcal{C}_r$ (Wagner and Yanai 2018; Xie et al. 2019; Chen et al. 2021). However, this assumption is difficult to satisfy for data in the wild. To take into account a more realistic situation, many researchers have begun to settle the open-set scenario, that is $\mathcal{C}_r \subset \mathcal{C}_t$ (Kimmel and Kelley 2020; Lotfollahi et al. 2022; Chen et al. 2022a,b). For ease of understanding, we define the cell types shared by the reference and target data as seen cell types and the cell types that only exist in the target data, but not in the reference data, as novel cell types.

Under the open-set assumption, many methods have been continuously proposed to solve this cell annotation problem, setting a goal whereby target cells are either labeled with seen cell types or uniformly classified into an "unassigned" group (De Kanter et al. 2019; Xu et al. 2021). Although they have achieved remarkable progress, simply annotating cells from novel cell types with an "unassigned" label is not conducive to subsequent downstream analysis, and it is universally vital to cluster them according to different cell types. Besides, the distribution shift of gene expression between reference and target data, namely batch effect, also affects annotation accuracy of the model (Shaham et al. 2017; Wang et al. 2019b; Lakkis et al. 2021). Thus, this paper focuses on a realistic and challenging scenario for cell annotation called generalized cell type annotation and discovery where cells in novel cell types are given cluster labels instead of "unassigned." Naturally, it can be argued that we could use the annotation methods for an open-set scenario to find "unassigned" cells first and then use clustering methods to divide them into groups. However, we show that such a two-step approach does not work well in practice. Therefore, it is necessary to develop a new annotation strategy dedicated to addressing this new setting in an end-to-end framework.

Here, we propose a novel method called scGAD for Generalized cell type Annotation and Discovery. It efficiently achieves label transfer over seen cell types and group packing of novel cell types. First, we build the intrinsic correspondences on seen and novel cell types by retrieving geometrically and semantically mutual nearest neighbors as anchor pairs. To reduce the potential negative impact of incorrectly identified anchors, we introduce a connection-aware attention mechanism to weigh the supervision of anchors. To transfer the known label information from reference data to target data and aggregate the new semantic knowledge only in the target data, we design a soft anchor-based self-

*These authors contributed equally.
†Corresponding Author.

supervised learning module to supervise the discriminative training on target data. Considering the compactness and separability of the embedding space, we propose a confidential prototype self-supervised learning paradigm to implicitly capture the consistent category structure of reference and target data. To evaluate the performance of scGAD fairly, we select various comparison baselines and carefully construct the single-data and cross-data benchmarks based on massive, highly imbalanced scRNA-seq data. Finally, by leveraging the clustering accuracy on reference data, we propose a solution to address the challenging and heretofore poorly investigated problem in single-cell annotation: estimating the number of cell types in unlabeled data.

We highlight the main contribution as follows:

- We propose a new, realistic, and challenging task called generalized cell type annotation and discovery in the single-cell annotation field. To effectively tackle this task, we further propose a novel method named scGAD.
- We introduce an anchor-based self-supervised learning module with a confidential prototypical self-supervised learning paradigm to achieve seen cell type annotation and novel cell type clustering simultaneously.
- We propose an easy, yet effective, solution to the challenging problem of estimating the total cell type number in target data.
- We design the comprehensive comparison baselines and evaluation benchmarks to validate the practicality of scGAD, and deeper analyses show the effectiveness of its proposed individual components.

Related Work

Single-Cell RNA-Seq Data Clustering

Recently, deep learning technologies have been applied to analyze scRNA-seq data (Talwar et al. 2018; Wang et al. 2019a; Arisdakessian et al. 2019). These technologies exhibit superior performance over traditional machine learning algorithms in various learning scenarios (Mereu et al. 2020; Flores et al. 2022). For example, scziDesk (Chen et al. 2020a) introduces a soft self-training k-means algorithm to cluster the cell population in the latent space, and it can correct the bias caused by hard clustering techniques. Also, scCNC (Wang et al. 2022) proposes a semi-supervised cell clustering method based on a capsule network which integrates domain knowledge into the clustering process. Finally, scNAME (Wan, Chen, and Deng 2022) designs a mask estimation task and a neighborhood contrastive learning framework to better exploit the cell's intrinsic structure and detect rare cell types. However, although these methods can discover the novel types in target data, they cannot recognize seen cell types that previously exist in reference data.

Single-Cell RNA-Seq Data Annotation

In addition to cell clustering, annotating each cell with the corresponding true cell type is also critical for downstream analysis. Traditional cell annotation involves searching for marker genes in each cluster (Vieth et al. 2019), which is not an accessible task for non-biologists. As more and more annotated scRNA-seq data become available, many research teams have turned to automatic annotation methods to simplify this step. Based on deep transfer learning, ItClust (Hu et al. 2020) utilizes cell-type-specific gene expression information learned from reference data to annotate target data. MARS (Brbić et al. 2020) introduces a meta-learning framework that can obtain cell-type knowledge by identifying commonality in the meta-dataset. scNym (Kimmel and Kelley 2020) integrates gene expression knowledge from reference and target data by applying a semi-supervised, adversarial learning technique. scArches (Lotfollahi et al. 2022) proposes a transfer learning and parameter fine-tuning strategy to leverage conditional neural network models adapted to target data. Overall, these methods can only roughly label cells from novel cell types with "unassigned" label, which is not conducive to subsequent downstream analysis.

Method

We first give some notations. We have access to a labeled reference data $\mathcal{D}_r = \{(x_i^r, y_i^r)_{i=1}^{n_r}\}$ and unlabeled target data $\mathcal{D}_t = \{(x_i^t, y_i^t)_{i=1}^{n_t}\}$, which can come from the same scRNA-seq dataset or different scRNA-seq datasets. Moreover, the label sets of reference data and target data are denoted as $\mathcal{C}_r$ and $\mathcal{C}_t$, respectively. In our problem, we assume that $\mathcal{C}_r \subset \mathcal{C}_t$; furthermore, the seen label set is defined as $\mathcal{C}_s = \mathcal{C}_r \cap \mathcal{C}_t$, and the novel label set is defined as $\mathcal{C}_n = \mathcal{C}_t \backslash \mathcal{C}_r$. To meet our goal, we either assign cell type labels of reference data or clustering labels to cells in the target data.

Considering the traits of scRNA-seq data, we assume that $\{x_i\}_{i=1}^{n_r+n_t}$ follows zero-inflated negative binomial distribution, and we use an autoencoder model to denoise data (Eraslan et al. 2019). Inspired by the recent progress in self-supervised learning (He et al. 2020; Chen et al. 2020d, 2022c), we use a data augmentation strategy to generate different views of gene expression in order to better capture the correlations across genes. The detailed information can be seen in the Supplementary Information (SI). In order to assign a classification or clustering label for each cell, we attach a classifier Φ to the latent layer, thereby mapping the embedding feature z_i to one of the $\mathcal{C}_r \cup \mathcal{C}_t$ cell types together with a predictive probability vector p_i (see Figure 1). Without loss of generality, we assume that the first $\mathcal{C}_s$ heads correspond to the seen cell types while the remaining $\mathcal{C}_n$ heads correspond to the novel cell types. The number of $\mathcal{C}_n$ can be estimated and entered into the model as known information. The specific estimation method will be introduced later. Since we also take the augmented data matrix as input, its embedding representation and predictive probability can be written as $\tilde{z}_i$ and $\tilde{p}_i$, respectively.

Anchor-Based Self-Supervised Learning

Based on the known labels of reference data, we can directly use the standard cross-entropy loss to train the classifier Φ on the reference data,

$$L_{sce} = -\frac{1}{2n_r} \sum_{i=1}^{n_r} \sum_{j=1}^{|\mathcal{C}_r \cup \mathcal{C}_t|} (y_{ij}^r \log p_{ij}^r + y_{ij}^r \log \tilde{p}_{ij}^r) \quad (1)$$

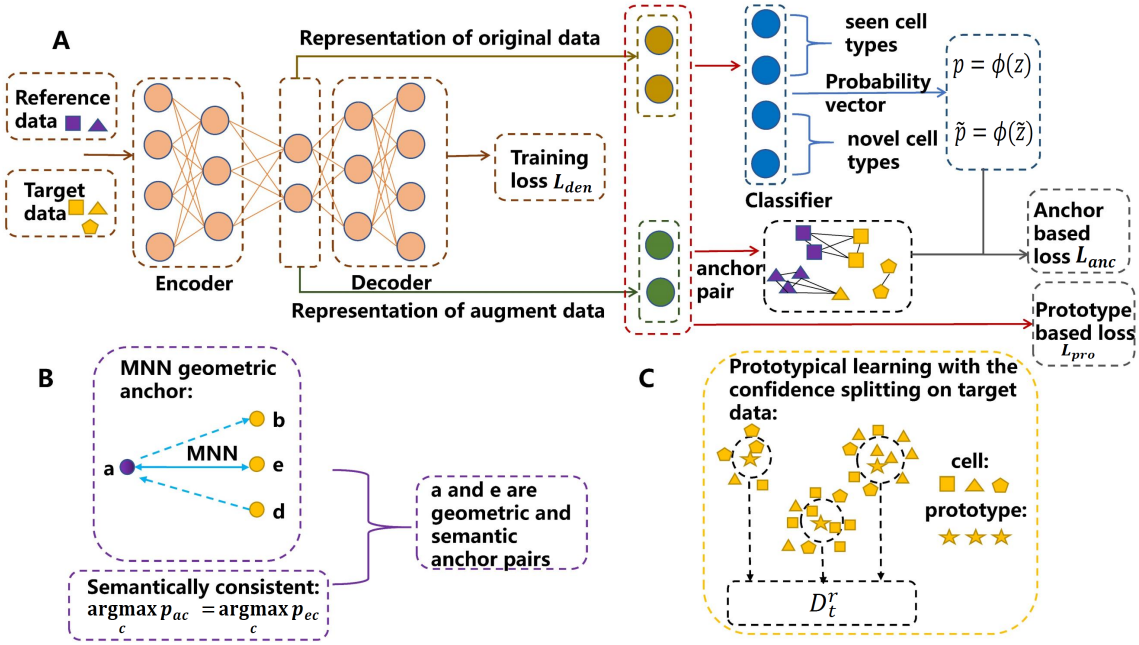


Figure 1: Schematics of scGAD. (A) Overall model consists of an autoencoder and a classifier. Anchor-based loss L_{anc} and Prototype-based loss L_{pro} are designed to train the model. (B) The definition of geometic and semantic anchor pair. (C) We add confidence splitting procedure into prototypical learning to avoid error propagation.

where y_i^r is written as the $|\mathcal{C}_r \cup \mathcal{C}_n|$-dimensional one-hot vector to facilitate calculations.

Training the classifier only on seen cell types without label supervision for novel cell types would lead to a prediction imbalance between two kinds of cell types, thus directly causing the model to misclassify novel cells into seen cell types. However, even though the target data possess the category shift, cells from the same cell types are still expected to be geometrically and semantically close to each other based on the manifold assumption (Lin and Zha 2022). Thus, we propose to align each target sample to the intimate anchors in reference data and target data.

Here, we introduce mutual nearest neighbors (MNN) as the geometric anchors between the reference and target data. Assume that the embedding features of reference and target data are $\{z_i^r\}_{i=1}^{n_r}$ and $\{z_i^t\}_{i=1}^{n_t}$, respectively. Then, for every cell z_i^r, we find g closest cells to cell z_i^r in Euclidean distance from $\mathcal{D}_t$ and denote this set as N_g^i. Similarly, we can construct M_g^j; that is, find g closest cells to z_j^t from $\mathcal{D}_r$. For a pair of cells (z_i^r, z_j^t), assuming they are in each other's neighborhood, namely $z_i^r \in M_g^j \wedge z_j^t \in N_g^i$, we call such pair an anchor pair between $\mathcal{D}_r$ and $\mathcal{D}_t$. However, the above definition of anchor pair is only based on the geometric distance in the embedding space and ignores the semantic information of cells expressed by the classifier, which may lead to negative knowledge transfer. Therefore, we further restrict the anchor pair semantically consistent, namely

$$z_i^r \in M_g^j \wedge z_j^t \in N_g^i \;\&\; \arg\max_l p_{il}^r = \arg\max_l p_{jl}^t, \quad (2)$$

where $1 \le l \le |\mathcal{C}_r \cup \mathcal{C}_t|$.

In order to represent the anchor pairs between $\mathcal{D}_r$ and $\mathcal{D}_t$ as a whole, we use $A_{rt} = A'_{tr}$ to denote their anchor adjacency matrix, and $A_{rt}(i,j) = 1$ when (z_i^r, z_j^t) is an anchor pair, otherwise $A_{rt}(i,j) = 0$. Similarly, we can also define the anchor pairs within $\mathcal{D}_t$; that is, (z_i^t, z_j^t) is an anchor pair only if it meets the following condition:

$$z_i^t \in N_g^j \wedge z_j^t \in N_g^i \;\&\; \arg\max_l p_{il}^t = \arg\max_l p_{jl}^t. \quad (3)$$

Define A_{tt} as the anchor adjacency matrix within $\mathcal{D}_t$, and we have $A_{tt} = 1$ if and only if (z_i^t, z_j^t) is an anchor pair, otherwise $A_{tt} = 0$. Although the anchor adjacency matrix A_{rr} within $\mathcal{D}_r$ can be directly obtained by the labels of reference samples, the information asymmetry may lead to an imbalance in the prediction of anchor pairs, so we still add geometric constraints on A_{rr}, namely

$$A_{rr}(i,j) = \begin{cases} 1 & z_i^r \in M_g^j \wedge z_j^r \in M_g^i \;\&\; y_i^r = y_j^r, \\ 0 & otherwise, \end{cases} \quad (4)$$

where y_i^r denotes the label of cell x_i^r and is known.

Robust identification of anchor pairs is essential for improving the discrimination on target data. To reduce the influence of false anchor pairs on prediction, we normalize the anchor adjacency matrix with a connectivity-aware weight. First of all, we concatenate different kinds of anchor adjacency matrices together to get the global adjacency matrix:

$$A = \begin{pmatrix} A_{rr} & A_{rt} \\ A_{tr} & A_{tt} \end{pmatrix}. \quad (5)$$

Let r_i be the degree of the i-th cell, i.e., the number of anchor pairs where the i-th cell is located, and then we can get a diagonal matrix $R = diag\{r_1, r_2, \cdots, r_{n_r+n_t}\}$. Thus, the normalized symmetric Laplace matrix W can be denoted as $W = R^{-\frac{1}{2}} A R^{-\frac{1}{2}} - I$, where $W(i,j) = \dfrac{A(i,j)}{\sqrt{r_i r_j}}$, $i \ne j$.

We regard $W(i,j)$ as the weighted affinity values of anchor pairs such that the larger the $W(i,j)$, the stronger the connectivity between the i-th cell and the j-th cell, thereby increasing reliability and, hence, believability that they belong to the same cell type. Similar to anchor adjacency matrix A, we can also write W in the form of a block matrix to facilitate the following discussion:

$$W = \begin{pmatrix} W_{rr} & W_{rt} \\ W_{tr} & W_{tt} \end{pmatrix}. \tag{6}$$

To propagate the known label message from $\mathcal{D}_r$ to $\mathcal{D}_t$ and aggregate the novel semantic knowledge within $\mathcal{D}_t$ itself, we use the prediction consistency between anchor pairs to supervise the discrimination training on $\mathcal{D}_t$. We then apply the affinity values of anchor pairs as attention weights. The anchor-guided self-supervised learning objective function can be given as

$$L_{ssl} = -\frac{1}{2n_t}\sum_{i=1}^{n_t}\Big[\sum_{\{j:W_{tr}(i,j)>0\}} W_{tr}(i,j)(p_j^{r\prime}p_i^t + \tilde{p}_j^{r\prime}\tilde{p}_i^t) + \sum_{\{j:W_{tt}(i,j)>0\}} W_{tt}(i,j)(p_j^{t\prime}p_i^t + \tilde{p}_j^{t\prime}\tilde{p}_i^t)\Big]. \tag{7}$$

With the affinity value as weight, we push prediction consistent to their associated anchors with strong connectivity and, to a lesser degree, to those with weak connectivity. As stated above, we can get the training loss function in the discrimination space, namely

$$L_{anc} = L_{sce} + L_{ssl}. \tag{8}$$

Confidential Prototypical Self-Supervised Learning

The prediction consistency constraint on matched anchors can be regarded as a cellular level self-supervised learning regularization. Despite alleviating the issue of imbalanced discriminative states, cell annotation may still encounter some potential risks. First, since the cells are classified at the cellular level, individual deviating cells can have a severe impact on the overall result; that is, our model may be sensitive to abnormal samples. Second, since weights of the classification head for novel cell types could be arbitrary, the loss function L_{ssl} may not be able to accurately separate seen and novel cell types in target data. Therefore, we propose to perform confidential prototypical self-supervised learning to implicitly encode the consensus category structure of reference and target data in the embedding space. Compared to cellular level alignment, matching a sample to a prototype is more robust to abnormal samples and makes the optimization converge faster and smoother.

In order to perform prototypical self-supervised learning on $\mathcal{D}_t$, we first perform k-means clustering on $\{z_i^t\}_{i=1}^{n_t}$ in the embedding space to obtain the corresponding class prototypes $\{\mu_i^t\}_{i=1}^{|\mathcal{C}_r\cup\mathcal{C}_t|}$ and the clustering labels $\{c_i\}_{i=1}^{n_t}$. Since the embedding features are not fully discriminative in the early stages of training, the clustering labels are very noisy. Relying on them to supervise model training can easily lead to error accumulation. Therefore, to avoid error propagation, we introduce a normalized confidence score to divide the target data into reliable set $\mathcal{D}_t^r$ and fuzzy set $\mathcal{D}_t^f$. This score is obtained by calculating the ratio of the distance between the sample and its own cluster center and the distance between the sample and the nearest non-self cluster center. The smaller the score, the more reliable the clustering label of the sample. In each training epoch, we default to select the top $\alpha\%$ samples with the lowest score in each cluster into $\mathcal{D}_t^r$, otherwise into $\mathcal{D}_t^f$. Then for samples in $\mathcal{D}_t^r$, we use their clustering labels to supervise the learning of their representations, while for samples in $\mathcal{D}_t^f$, we perform assignment entropy minimization to move them near the intimate prototypes. Specifically, we use the t-distribution kernel function to measure the similarity between each cell and the prototypes. Assuming that q_{ij}^t, $\tilde{q}_{ij}^t$ represents the affinity between the i-th prototype and the j-th cell, we have

$$q_{ij}^t = \frac{(1+||z_i^t-\mu_j^t||_2^2)^{-1}}{\sum_{l=1}^{|\mathcal{C}_r\cup\mathcal{C}_t|}(1+||z_i^t-\mu_l^t||_2^2)^{-1}}, \tag{9}$$

$$\tilde{q}_{ij}^t = \frac{(1+||\tilde{z}_i^t-\mu_j^t||_2^2)^{-1}}{\sum_{l=1}^{|\mathcal{C}_r\cup\mathcal{C}_t|}(1+||\tilde{z}_i^t-\mu_l^t||_2^2)^{-1}}. \tag{10}$$

Then the within $\mathcal{D}_t$ prototypical self-supervised learning objective can be written as

$$L_{pro}^t = -\frac{1}{2n_t}\sum_{i=1}^{n_t}[(\log q_{ic_i}^t + \log\tilde{q}_{ic_i}^t)I_{i\in\mathcal{D}_t^r} + \sum_{j=1}^{|\mathcal{C}_r\cup\mathcal{C}_t|}(q_{ij}^t\log q_{ij}^t + \tilde{q}_{ij}^t\log\tilde{q}_{ij}^t)I_{i\in\mathcal{D}_t^f}]. \tag{11}$$

By minimizing L_{pro}^t, the model can improve the compactness within clusters and the separability between clusters in the embedding space. Moreover, using L_{pro}^t as regularization can effectively satisfy the issue of blurred classification boundaries across cell types that may be caused by the connection of paired samples.

To better utilize the label knowledge of reference data to assist in prototype learning, we consider performing cell-prototype dual alignment from reference data to target prototypes to enforce learning category-aligned and discriminative embedding features. Specifically, we discover the positive matching, as well as negative matching, between labeled cells and cluster prototypes across $\mathcal{D}_r$ and $\mathcal{D}_t$. Similarly, q_i^r and $\tilde{q}_i^r$, the similarity distribution vectors on $\mathcal{D}_r$, can be computed as

$$q_{ij}^r = \frac{(1+||z_i^r-\mu_j^t||_2^2)^{-1}}{\sum_{l=1}^{|\mathcal{C}_r\cup\mathcal{C}_t|}(1+||z_i^r-\mu_l^t||_2^2)^{-1}}, \tag{12}$$

$$\tilde{q}_{ij}^r = \frac{(1+||\tilde{z}_i^r-\mu_j^t||_2^2)^{-1}}{\sum_{l=1}^{|\mathcal{C}_r\cup\mathcal{C}_t|}(1+||\tilde{z}_i^r-\mu_l^t||_2^2)^{-1}}. \tag{13}$$

Considering that reference samples come with accurate cell types, we naturally anticipate that the similarity distributions

in the same labels will be consistent. Thus, our cross-data prototypical self-supervised objective for $\mathcal{D}_r$ is defined as

$$L_{pro}^r = -\frac{1}{2n_r}\sum_{i=1}^{n_r}\sum_{j=1}^{|\mathcal{C}_r\cup\mathcal{C}_t|}(q_{ij}^r \log q_{ij}^r + \tilde{q}_{ij}^r \log \tilde{q}_{ij}^r) \quad (14)$$
$$+ \frac{1}{n_r^2}\sum_{i=1}^{n_r}\sum_{s=1}^{n_r}\|q_i^r - \tilde{q}_s^r\|_2^2 I_{\{y_i^r=y_s^r\}}.$$

By minimizing L_{pro}^r, the alignment between reference samples and target prototypes can be achieved. Such dual alignment is also conducive to better information exchange. Combining L_{pro}^t with L_{pro}^r, we give the total prototypical self-supervised learning loss as

$$L_{pro} = L_{pro}^t + L_{pro}^r. \quad (15)$$

Lastly, together with the data denoising loss L_{den} (see SI), we give the overall training objective as

$$L_{tol} = L_{den} + L_{anc} + L_{pro}. \quad (16)$$

Estimation of the $|\mathcal{C}_t|$ Value

Here, we propose a solution to a challenging and underinvestigated problem in cell annotation: estimating the cell type number $|\mathcal{C}_t|$ in target data. Almost all annotation methods assume that the number of $|\mathcal{C}_t|$ is a prior. However, this assumption is unrealistic in the real world. This calls on the community to develop a method for estimating $|\mathcal{C}_t|$. Our main idea derives from the information available in $\mathcal{D}_r$. Specifically, we perform k-means clustering on the whole dataset $\mathcal{D}_r \bigcup \mathcal{D}_t$ and then evaluate clustering accuracy only on the reference data $\mathcal{D}_r$. Let $|\hat{\mathcal{C}}_t|$ represent the estimated value. If $|\hat{\mathcal{C}}_t| > |\mathcal{C}_t|$, then $\hat{\mathcal{C}}_t - \mathcal{C}_t$ can be called the extra cell types, and all cells assigned to extra cell types are mispredicted. Similarly, if $|\hat{\mathcal{C}}_t| < |\mathcal{C}_t|$, then $\mathcal{C}_t - \hat{\mathcal{C}}_t$ can be called the extra true cell types, and all cells with those cell types are predicted incorrectly. Based on this analysis, whether $|\hat{\mathcal{C}}_t|$ is higher or lower than $|\mathcal{C}_t|$ will have a negative impact on the clustering accuracy on $\mathcal{D}_r$. In other words, the clustering accuracy on $\mathcal{D}_r$ will be maximized when $|\hat{\mathcal{C}}_t| = |\mathcal{C}_t|$. According to this intuition, we use $AC = f(|\hat{\mathcal{C}}_t|, \mathcal{D}_r)$ to measure the clustering accuracy on $\mathcal{D}_r$, which we optimize with Brent's algorithm to find the optimal $|\hat{\mathcal{C}}_t|$ (Brent 2013).

Experiment

Setup

Dataset. Our experiment consists of two parts: intra-data annotation and cross-data annotation. For the former, we collect 10 datasets sequenced from different organisms and platforms. The cell numbers range from 6462 to 110704, and the cell type numbers vary from 9 to 45. Unless otherwise noted, we first divide all cell types into 50% seen and 50% novel. Then we select 50% samples in seen cell types as $\mathcal{D}_r$ and the rest as $\mathcal{D}_t$. For the latter, we select 10 groups of datasets. Each group consists of a reference dataset and a target dataset, and batch effect exists between them. Their cell numbers range from a few thousand to tens of thousands, and the cell type number in the reference data is nearly half that of the target data. The basic information of these datasets can be seen in SI.

Baselines. Our task is to establish a new cell annotation setting for which no ready-to-use baselines exist. Thus, we compare scGAD with recently developed scRNA-seq clustering and annotation algorithms, including three clustering methods (scziDesk, scCNC, and scNAME) and four annotation methods (MARS, ItClust, scNym, and scArches). For clustering methods, only scCNC participates in training with both $\mathcal{D}_r$ and $\mathcal{D}_t$, while the other two train only on $\mathcal{D}_t$. We report their clustering performance on seen and novel cell types. For annotation methods, we first use them to classify target cells into seen cell types and identify the "unassigned" group. Next, we apply k-means clustering on the "unassigned" group to obtain novel clusters. The detailed information of these baselines can be seen in SI.

Evaluation protocols. We report the classification accuracy on seen cell types and clustering accuracy on novel cell types for scGAD and annotation baselines, while report clustering accuracy on both seen and novel cell types for clustering baselines. Specifically, to compute the clustering accuracy, we apply the Hungarian algorithm to solve the optimal assignment problem (Kuhn 1955). When reporting accuracy on all cell types, we solve the optimal assignment problem on both seen and novel cell types. The reported accuracies are the mean values of three runs.

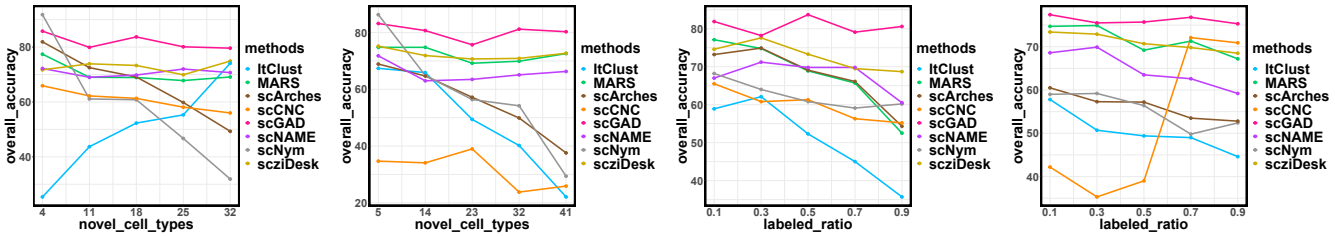
Implementation details. Our algorithm is implemented by PyTorch, and we conduct the experiments with 2 Tesla A100 GPUs. The two layers of the encoder are sized as 512 and 256, respectively, and the decoder has the reverse structure of the encoder. The bottleneck layer has a size of 128. The training mini-batch size is set to 256, and the optimizer is Adam with learning rate 1e-4. The neighbor number g is set to 10, and the quantile α is set to 20. We first pretrain the whole model using L_{den} loss with 600 epochs. Then, we apply the standard k-means algorithm on target embedding features to obtain cluster centers as the initial values of prototypes. Finally, we train the model with the overall loss L_{tol} until the predictions no longer change.

Results Comparison

Intra-data annotation. We begin by exploring the intra-data annotation scenario without batch effect, and Table 1 summarizes the performance of scGAD and other baselines on ten datasets. Overall, scGAD performs better than other methods, indicating the effectiveness of our strategy. Especially, scGAD beats other baselines on both annotation accuracy and clustering accuracy, thereby validating the superiority of it on this new setting. In other words, the results indicate that the two-step methods of first classifying and then clustering can only provide suboptimal results. For example, scNym can always achieve impressive results in annotation accuracy, even surpassing scGAD on individual datasets. Such result is not surprising since scNym prefers to recognize some confusing cells as seen cell types, and the catastrophic failure of scNym in clustering accuracy is enough to illustrate this point. As a competitive annotation method, MARS can also annotate novel cells with clustering

	Cao			Hochane			Park			Quake 10x			Quake Smart-seq2		
	seen	novel	overall	seen	novel	overall	seen	novel	overall	seen	novel	overall	seen	novel	overall
scziDesk	85.2	74.1	63.8	91.0	83.9	84.4	97.3	72.9	85.0	84.1	58.5	73.3	76.7	72.5	70.7
scNAME	79.1	78.5	75.1	91.0	85.7	84.3	56.6	79.5	73.4	82.2	62.0	69.8	76.5	61.2	63.5
scCNC	50.2	60.9	52.7	94.1	70.0	70.4	92.4	61.0	76.6	85.0	49.8	61.3	65.0	40.8	39.0
MARS	88.6	75.8	64.3	96.9	74.5	78.8	61.6	78.2	68.3	92.1	52.8	68.9	80.3	70.6	69.2
ItClust	14.5	62.3	56.6	33.1	49.5	45.3	76.3	42.6	62.4	70.5	47.3	52.3	32.7	55.5	49.4
scNym	**99.2**	69.4	66.2	98.9	49.8	46.0	**99.8**	48.9	45.2	**98.4**	52.8	60.8	**96.9**	59.2	56.4
scArches	73.4	46.5	52.2	82.6	91.5	**89.3**	86.6	36.8	65.7	88.3	56.6	69.1	72.3	54.7	57.2
scGAD	92.4	**81.0**	**78.3**	**99.5**	**93.8**	84.9	97.7	**80.9**	**88.4**	95.8	**62.1**	**83.7**	91.3	**76.3**	**75.7**
	Wagner			**Zeisel**			**Zheng**			**Chen**			**Guo**		
	seen	novel	overall	seen	novel	overall	seen	novel	overall	seen	novel	overall	seen	novel	overall
scziDesk	72.1	48.2	54.6	78.0	89.1	82.5	57.7	52.0	45.7	78.8	92.2	90.8	99.6	80.0	75.1
scNAME	74.4	48.4	54.8	93.4	88.1	84.3	57.7	52.0	45.7	79.8	92.1	91.1	99.8	76.4	72.0
scCNC	85.8	51.4	55.0	77.2	50.8	55.0	61.5	56.6	48.6	79.8	92.1	91.1	99.8	76.4	72.0
MARS	81.6	42.6	50.9	98.9	83.3	84.1	72.5	59.5	50.6	80.1	**94.1**	90.4	99.7	72.6	68.8
ItClust	18.5	32.2	36.4	52.7	57.3	54.1	20.7	50.9	43.8	20.8	82.5	72.2	19.5	74.7	67.8
scNym	**96.5**	42.3	44.2	99.6	64.6	62.7	**98.8**	56.5	51.4	97.4	77.7	72.2	99.8	60.4	56.8
scArches	58.1	35.9	41.7	78.1	60.0	63.3	60.4	72.9	**68.4**	74.4	85.6	82.9	61.0	78.9	74.8
scGAD	92.1	**49.6**	**56.4**	**99.7**	**89.7**	**88.1**	97.6	**74.8**	66.3	**98.3**	93.0	**94.1**	**99.9**	**82.4**	**75.2**

Table 1: Performance comparison on ten real datasets in intra-data annotation experiments.



(a) Quake 10x (b) Quake Smart-seq2 (c) Quake 10x (d) Quake Smart-seq2

Figure 2: Accuracy on all cell types. (a, b) Changing novel cell type number in Quake 10x and Quake Smart-seq2 datasets, respectively; (c, d) Changing labeled ratio in Quake 10x and Quake Smart-seq2 datasets, respectively.

label, but scGAD still beats it with clear margins. This evidence again emphasizes the necessity of scGAD to balance the prediction states of cells in seen and novel cell types by introducing anchor pairs. The clustering methods scziDesk and scNAME do not utilize the label information of the reference data and thus lose competitiveness in annotation accuracy on seen cell types. Furthermore, they can only label each cluster with clustering label, rather than true cell type label, which is a common limitation with clustering methods in the real-world scenario. In summary, we can conclude that scGAD outperforms other baselines on three evaluation indexes in the intra-data annotation task.

Cross-data annotation. Next, we study the performance of each method on the cross-data annotation scenario with batch effect. As shown in Table 2, the batch effect does, indeed, have a certain influence on the accuracy of all methods. This is reasonable because batch effect makes cells of the same cell type move away from each other, thus leading to a more challenging annotation task. Even so, scGAD still achieves an impressive performance that is only slightly affected by the batch effect, indicating that our confidential prototypical self-supervised learning strategy removes batch effect to a certain extent implicitly. However, the performance of other methods is severely degraded. Especially, MARS and Itclust separate reference and target data in training, elevating susceptibility to batch effect and, in turn, leading to model overfitting and false cell type annotations.

Ablation Study

Robustness Analysis. Since novel cell type number $|\mathcal{C}_n|$ determines the difficulty of clustering novel cells, it is imperative to explore the impact of $|\mathcal{C}_n|$. We chose to conduct experiments on Quake 10x and Quake Smart-seq2 datasets, which have the most cell types. Here, $|\mathcal{C}_n|$ varies in the range of $[4, 11, 18, 25, 32]$ for Quake 10x and $[5, 14, 23, 32, 41]$ for Quake Smart-seq2. For ease of illustration, the results are shown in Figure 2(a) and 2(b) in the form of line graphs. No matter what value $|\mathcal{C}_n|$ takes, it is easy to see that scGAD always performs better than other baselines, validating its superiority. Moreover, the polyline of scGAD is smoother, which demonstrates its robustness. The overall accuracy of scNym, scArches and scCNC drops catastrophically with increasing $|\mathcal{C}_n|$ which is reasonable because they focus more on annotating cells of seen cell types, and increasing $|\mathcal{C}_n|$ will give them more interference. The result for ItClust rises dramatically on Quake 10x and drops significantly on Quake Smart-seq2, indicating its instability. MARS, scziDesk and scNAME are relatively stable for the variety of $|\mathcal{C}_n|$, but their overall accuracy is lower than that of scGAD. As stated above, we can conclude that scGAD is more robust and sta-

	Enge (R) Baron_human (T)			Lawlor (R) Baron_human (T)			Muraro (R) Baron_human (T)			Xin (R) Baron_human (T)			Vento _10x (R) Vento_Smart-seq2 (T)		
	seen	novel	overall	seen	novel	overall	seen	novel	overall	seen	novel	overall	seen	novel	overall
scziDesk	81.6	81.5	81.6	81.3	80.3	81.2	81.6	81.5	81.6	75.1	84.2	81.3	81.7	79.0	81.5
scNAME	81.0	81.9	81.2	80.7	79.4	79.9	95.8	71.4	91.2	73.6	85.3	77.7	87.4	80.3	86.0
scCNC	47.6	38.5	38.8	54.0	43.9	40.9	75.0	40.8	61.1	46.6	54.7	36.5	92.1	63.4	84.8
MARS	90.3	**86.2**	79.8	80.9	**90.7**	80.3	79.5	82.3	80.0	93.6	78.0	88.6	71.3	78.6	70.3
ItClust	83.4	52.3	72.7	88.5	48.9	77.1	80.9	56.4	69.2	84.5	80.7	84.1	79.8	50.7	70.4
scNym	**97.7**	71.9	84.7	90.2	52.2	82.8	88.2	55.5	63.9	**97.9**	40.0	52.3	98.7	66.5	75.9
scArches	89.2	58.0	80.3	47.3	66.8	52.5	89.3	52.8	80.9	61.5	52.2	52.7	87.6	52.9	78.2
scGAD	96.3	82.6	**93.8**	**96.6**	82.6	**90.3**	**96.5**	**83.2**	**93.7**	93.6	**86.0**	**91.3**	**98.8**	**80.5**	**92.4**

	Vento Smart-seq2 (R) Vento 10x (T)			Plasschaert (R) Montoro 10x (T)			M Smart-seq2 (R) M 10x (T)			Haber largecell (R) Haber region (T)			Haber region (R) Haber largecell (T)		
	seen	novel	overall	seen	novel	overall	seen	novel	overall	seen	novel	overall	seen	novel	overall
scziDesk	88.4	98.4	90.9	67.9	74.6	68.3	94.0	89.3	91.2	43.9	60.9	53.0	85.3	80.8	71.0
scNAME	86.5	98.2	92.8	95.1	90.2	96.0	93.7	99.0	96.8	46.0	62.6	54.3	89.1	80.9	71.6
scCNC	83.4	47.1	43.7	79.7	73.1	73.0	92.4	65.5	76.2	62.7	69.4	55.9	75.7	50.4	51.6
MARS	94.5	78.6	83.8	88.6	**94.5**	89.1	81.5	97.5	86.9	57.1	75.1	68.2	83.8	64.1	67.1
ItClust	64.3	75.0	58.2	90.1	75.1	83.2	36.8	70.5	67.2	53.4	58.2	56.4	6.2	64.5	53.6
scNym	98.1	70.4	80.6	**96.1**	77.7	83.1	**95.1**	48.6	49.8	**95.8**	44.4	51.2	84.2	53.7	53.0
scArches	83.4	66.8	75.2	91.4	67.4	85.3	62.0	55.5	59.0	72.3	51.7	59.6	71.9	45.4	50.4
scGAD	**98.4**	**99.2**	**97.4**	93.6	94.0	**96.2**	94.1	**99.1**	**97.1**	86.2	**97.4**	**93.6**	**89.8**	**81.5**	**72.2**

Table 2: Performance comparison in cross-data annotation experiments. "R": reference data; "T": target data. Specially, M Smart-seq2 and M 10x stand for Mammary Smart-seq2 and Mammary 10x respectively.

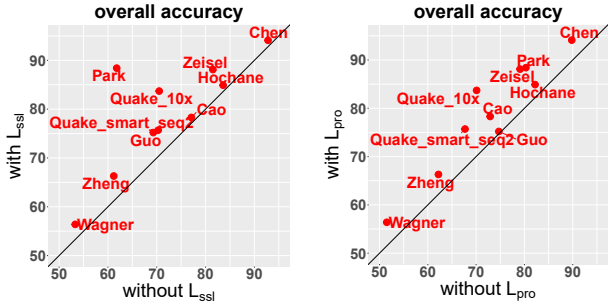
increment	-12	-6	0	6	12
Quake 10x	92.2	94.7	95.1	94.9	93.6
Quake Smart-seq2	78.1	90.5	91.6	91.2	90.1

Table 3: Clustering accuracy on seen cell types when vary $|\mathcal{C}_t|$-value on Quake 10x and Quake Smart-seq2 datasets.

ble than other baselines.

Since the ratio of labeled data determines how much information can be provided from the reference dataset, we also explore its impact by conducting experiments on Quake 10x and Quake Smart-seq2 datasets. Figures 2(c) and 2(d) show the line graphs of overall accuracy with the change in the ratio of labeled data, which varies in the range of $[0.1, 0.3, 0.5, 0.7, 0.9]$. We find that scGAD still achieves consistently better results than the other baselines and can maintain its excellent performance without being affected by the ratio of labeled data. However, the other methods are affected by the varied ratio of labeled data and tendency toward providing suboptimal results. In conclusion, scGAD can provide reliable and remarkable performance, even with a few labeled data. Because of limited space, we put other experimental results in SI.

Validity of the $|\mathcal{C}_t|$ value estimation method. The $|\mathcal{C}_t|$ value represents the cell type number in target data, and its estimation plays an important role in discovering novel cell types. To verify the validity of our estimation method, we conduct experiments on Quake 10x and Quake Smart-seq2 datasets, which have 36 and 45 cell types, respectively. We study the case where the increment of $|\mathcal{C}_t|$ varies in the range of $[-12, -6, 0, 6, 12]$, and increment=0 means that the estimation of $|\mathcal{C}_t|$ gets the true value. The results on these two datasets are shown in Table 3. We see that the accuracy rises to maximum value at increment=0, regardless of the datasets, indicating the validity of our estimation method.



(a) w/o $\mathcal{L}_{ssl}$ (b) w/o $\mathcal{L}_{pro}$

Figure 3: Ablation Study. (a, b) Comparing the accuracy on all cell types with or without $\mathcal{L}_{ssl}$ and $\mathcal{L}_{pro}$, respectively.

Effect of $\mathcal{L}_{ssl}$ and $\mathcal{L}_{pro}$. Here, we study the contribution of different components in scGAD on ten datasets by removing the anchor-guided self-supervised learning module $\mathcal{L}_{ssl}$ and the confidential prototypical self-supervised learning module $\mathcal{L}_{pro}$. The results are shown in Figure 3. When $\mathcal{L}_{ssl}$ or $\mathcal{L}_{pro}$ is dropped, we can see that the performance of scGAD decreases significantly for all datasets, which fully verifies the roles of $\mathcal{L}_{ssl}$ and $\mathcal{L}_{pro}$ in the whole framework.

Conclusion

In this paper, we introduce a new and practical generalized cell type annotation and discovery task for scRNA-seq data and propose a novel algorithm called scGAD to address it. In scGAD, we design two effective modules, namely weighted anchor-based self-supervised learning framework and confidential prototypical self-supervised paradigm. To evaluate the algorithm's performance, we construct comprehensive baselines and benchmarks. Extensive results verify the superiority and robustness of scGAD compared to competing annotation and clustering methods.

References

Arisdakessian, C.; Poirion, O.; Yunits, B.; Zhu, X.; and Garmire, L. X. 2019. DeepImpute: an accurate, fast, and scalable deep neural network method to impute single-cell RNA-seq data. *Genome biology*, 20(1): 1–14.

Brbić, M.; Zitnik, M.; Wang, S.; Pisco, A. O.; Altman, R. B.; Darmanis, S.; and Leskovec, J. 2020. MARS: discovering novel cell types across heterogeneous single-cell experiments. *Nature methods*, 17(12): 1200–1206.

Brent, R. P. 2013. *Algorithms for minimization without derivatives*. Courier Corporation.

Cao, Z.-J.; Wei, L.; Lu, S.; Yang, D.-C.; and Gao, G. 2019. Cell BLAST: searching large-scale scRNA-seq databases via unbiased cell embedding. *BioRxiv*, 587360.

Chen, L.; Du, Q.; Lou, Y.; He, J.; Bai, T.; and Deng, M. 2022a. Mutual nearest neighbor contrast and hybrid prototype self-training for universal domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 6248–6257.

Chen, L.; He, Q.; Zhai, Y.; and Deng, M. 2021. Single-cell RNA-seq data semi-supervised clustering and annotation via structural regularized domain adaptation. *Bioinformatics*, 37(6): 775–784.

Chen, L.; Lou, Y.; He, J.; Bai, T.; and Deng, M. 2022b. Evidential neighborhood contrastive learning for universal domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 6258–6267.

Chen, L.; Lou, Y.; He, J.; Bai, T.; and Deng, M. 2022c. Geometric anchor correspondence mining with uncertainty modeling for universal domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16134–16143.

Chen, L.; Wang, W.; Zhai, Y.; and Deng, M. 2020a. Deep soft K-means clustering with self-training for single-cell RNA sequence data. *NAR genomics and bioinformatics*, 2(2): lqaa039.

Chen, L.; Wang, W.; Zhai, Y.; and Deng, M. 2020b. Single-cell transcriptome data clustering via multinomial modeling and adaptive fuzzy k-means algorithm. *Frontiers in genetics*, 11: 295.

Chen, L.; Zhai, Y.; He, Q.; Wang, W.; and Deng, M. 2020c. Integrating deep supervised, self-supervised and unsupervised learning for single-cell RNA-seq clustering and annotation. *Genes*, 11(7): 792.

Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020d. A simple framework for contrastive learning of visual representations. *In International conference on machine learning*, 1597–1607.

De Kanter, J. K.; Lijnzaad, P.; Candelli, T.; Margaritis, T.; and Holstege, F. C. 2019. CHETAH: a selective, hierarchical cell type identification method for single-cell RNA sequencing. *Nucleic acids research*, 47(16): e95–e95.

Eraslan, G.; Simon, L. M.; Mircea, M.; Mueller, N. S.; and Theis, F. J. 2019. Single-cell RNA-seq denoising using a deep count autoencoder. *Nature communications*, (1): 1–14.

Flores, M.; Liu, Z.; Zhang, T.; Hasib, M. M.; Chiu, Y.-C.; Ye, Z.; and Paniagua, K. e. a. 2022. Deep learning tackles single-cell analysis—a survey of deep learning for scRNA-seq analysis. *Briefings in Bioinformatics*, 23(1): bbab531.

He, K.; Fan, H.; Wu, Y.; Xie, S.; and Girshick, R. 2020. Momentum contrast for unsupervised visual representation learning. *In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9729–9738.

Hu, J.; Li, X.; Hu, G.; Lyu, Y.; Susztak, K.; and Li, M. 2020. Iterative transfer learning with neural network for clustering and cell type classification in single-cell RNA-seq analysis. *Nature machine intelligence*, 2(10): 607–618.

Kimmel, J. C.; and Kelley, D. R. 2020. scNym: Semi-supervised adversarial neural networks for single cell classification. *bioRxiv*, 2020–06.

Kuhn, H. W. 1955. The Hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2): 83–97.

Lakkis, J.; Wang, D.; Zhang, Y.; Hu, G.; Wang, K.; Pan, H.; Ungar, L.; Reilly, M. P.; Li, X.; and Li, M. 2021. A joint deep learning model enables simultaneous batch effect correction, denoising, and clustering in single-cell transcriptomics. *Genome research*, 31(10): 1753–1766.

Lin, T.; and Zha, H. 2022. Riemannian manifold learning. *IEEE transactions on pattern analysis and machine intelligence*, 30(5).

Lotfollahi, M.; Naghipourfar, M.; Luecken, M. D.; Khajavi, M.; Büttner, M.; Wagenstetter, M.; Avsec, Ž.; Gayoso, A.; Yosef, N.; Interlandi, M.; et al. 2022. Mapping single-cell data to reference atlases by transfer learning. *Nature Biotechnology*, 40(1): 121–130.

Mereu, E.; Lafzi, A.; Moutinho, C.; Ziegenhain, C.; McCarthy, D. J.; Álvarez Varela, A.; and Batlle, E. e. a. 2020. Benchmarking single-cell RNA-sequencing protocols for cell atlas projects. *Nature biotechnology*, 38(6): 747–755.

Shaham, U.; Stanton, K. P.; Zhao, J.; Li, H.; Raddassi, K.; Montgomery, R.; and Kluger, Y. 2017. Removal of batch effects using distribution-matching residual networks. *Bioinformatics*, 33(16): 2539–2546.

Talwar, D.; Mongia, A.; Sengupta, D.; and Majumdar, A. 2018. AutoImpute: Autoencoder based imputation of single-cell RNA-seq data. *Scientific reports*, 8(1): 1–11.

Vieth, B.; Parekh, S.; Ziegenhain, C.; Enard, W.; and Hellmann, I. 2019. A systematic evaluation of single cell RNA-seq analysis pipelines. *Nature communications*, 10(1): 1–11.

Wagner, F.; and Yanai, I. 2018. Moana: a robust and scalable cell type classification framework for single-cell RNA-Seq data. *BioRxiv*, 456129.

Wan, H.; Chen, L.; and Deng, M. 2022. scNAME: neighborhood contrastive clustering with ancillary mask estimation for scRNA-seq data. *Bioinformatics*, 38(6): 1575–1583.

Wang, H.-Y.; Zhao, J.-P.; Zheng, C.-H.; and Su, Y.-S. 2022. scCNC: a method based on capsule network for clustering scRNA-seq data. *Bioinformatics*.

Wang, J.; Agarwal, D.; Huang, M.; Hu, G.; Zhou, Z.; Ye, C.; and Zhang, N. R. 2019a. Data denoising with transfer learning in single-cell transcriptomics. *Nature methods*, 16(9): 875–878.

Wang, T.; Johnson, T. S.; Shao, W.; Lu, Z.; Helm, B. R.; Zhang, J.; and Huang, K. 2019b. BERMUDA: a novel deep transfer learning method for single-cell RNA sequencing batch correction reveals hidden high-resolution cellular subtypes. *Genome biology*, 20(1): 1–15.

Xie, P.; Gao, M.; Wang, C.; Zhang, J.; Noel, P.; Yang, C.; Von Hoff, D.; Han, H.; Zhang, M. Q.; and Lin, W. 2019. SuperCT: a supervised-learning framework for enhanced characterization of single-cell transcriptomic profiles. *Nucleic acids research*, 47(8): e48–e48.

Xu, C.; Lopez, R.; Mehlman, E.; Regier, J.; Jordan, M. I.; and Yosef, N. 2021. Probabilistic harmonization and annotation of single-cell transcriptomics data with deep generative models. *Molecular systems biology*, (1): e9620.

Yuan, M.; Chen, L.; and Deng, M. 2022. scMRA: a robust deep learning method to annotate scRNA-seq data with multiple reference datasets. *Bioinformatics*, 38(3): 738–745.