Bioinformatics, 38(11), 2022, 2996–3003
https://doi.org/10.1093/bioinformatics/btac199
Advance Access Publication Date: 8 April 2022
Original Paper

Gene expression

scGraph: a graph neural network-based approach to automatically identify cell types

Qijin Yin [1], Qiao Liu [2], Zhuoran Fu[1], Wanwen Zeng[2,3], Boheng Zhang[1], Xuegong Zhang [1], Rui Jiang [1] and Hairong Lv[1,4,*]

[1]Ministry of Education Key Laboratory of Bioinformatics, Research Department of Bioinformatics at the Beijing National Research Center for Information Science and Technology, Center for Synthetic and Systems Biology, Department of Automation, Tsinghua University, Beijing 100084, China, [2]Department of Statistics, Stanford University, Stanford, CA 94305, USA, [3]College of Software, Nankai University, Tianjin 300350, China and [4]Fuzhou Institute of Data Technology, Fuzhou 350200, China

*To whom correspondence should be addressed.
Associate Editor: Anthony Mathelier

Received on September 22, 2021; revised on December 13, 2021; editorial decision on December 14, 2021; accepted on April 7, 2020

Abstract

Motivation: Single-cell technologies play a crucial role in revolutionizing biological research over the past decade, which strengthens our understanding in cell differentiation, development and regulation from a single-cell level perspective. Single-cell RNA sequencing (scRNA-seq) is one of the most common single cell technologies, which enables probing transcriptional states in thousands of cells in one experiment. Identification of cell types from scRNA-seq measurements is a fundamental and crucial question to answer. Most previous studies directly take gene expression as input while ignoring the comprehensive gene–gene interactions.

Results: We propose scGraph, an automatic cell identification algorithm leveraging gene interaction relationships to enhance the performance of the cell-type identification. scGraph is based on a graph neural network to aggregate the information of interacting genes. In a series of experiments, we demonstrate that scGraph is accurate and outperforms eight comparison methods in the task of cell-type identification. Moreover, scGraph automatically learns the gene interaction relationships from biological data and the pathway enrichment analysis shows consistent findings with previous analysis, providing insights on the analysis of regulatory mechanism.

Availability and implementation: scGraph is freely available at https://github.com/QijinYin/scGraph and https://figshare.com/articles/software/scGraph/17157743.

Contact: lvhairong@tsinghua.edu.cn

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Single-cell RNA sequencing (scRNA-seq) technology, which measured the embryo development state in a single mouse blastomere at first, has quickly developed and greatly promoted the understanding of biological science in the past decade. To date, the advantages of single-cell technologies have been largely extended for measuring high-resolution profile for thousands of individual cells in different respects, including DNA modification, chromatin accessibility state and gene expression (Mezger *et al.*, 2018), which gives us a great opportunity for studying the trans- and cis-regulatory mechanism in a single-cell resolution (Macaulay *et al.*, 2017). A large-scale of public single-cell data, especially scRNA-seq, has been rapidly accumulated. Although several different scRNA-seq protocols have been developed to measure transcriptome at a single-cell level, there still exists unavoidable technological defects, such as technical noise (e.g. batch effect), that might diminish the quality of the observed data. Nevertheless, scRNA-seq data contain abundant transcriptomic information, leading to a wide range of applications, such as cell-type identification (Abdelaal *et al.*, 2019; Ma and Pellegrini, 2020), cell development trajectory analysis (Saelens *et al.*, 2019) and gene regulatory analysis (Yuan and Bar-Joseph, 2019).

In single-cell analysis, dimension reduction and clustering are the most common and crucial computational tasks before downstream analysis, for which many approaches have been developed (Chen *et al.*, 2021a). Cell-type identification heavily relies on an optimal clustering result, which is highly subjective due to the lack of ground-truth labels. Furthermore, the clustering and labeling of cell types of a scRNA-seq dataset are heavily dependent on the expression of cluster-specific genes and the prior knowledge of different cell types, respectively, which requires comprehensive prior knowledge

of marker genes for each cell type (Pliner *et al.*, 2019). With the rapid accumulation of scRNA-seq data spanning across specific tissues, organs and even species, it is meaningful to build a computational model to automatically identify cell types for newly sequenced cells. Such a model can leverage information from these public datasets to determine the cell type for individual cells, thus eliminating the issue of subjectivity while subsequently reducing the complexity of the analysis workflow.

With plentiful annotated and publicly available scRNA-seq datasets, many computational methods have been developed. CHETAH (de Kanter *et al.*, 2019) is a cell-type identification algorithm that assigns cell types in a hierarchical manner by correlating the queried scRNA-seq data with references. scID (Boufea *et al.*, 2020) identifies transcriptionally related cell types among scRNA-seq datasets via a linear discriminant analysis framework. SingleR (Aran *et al.*, 2019) assigns cellular identity for scRNA-seq based on correlating gene expression between query scRNA-seq data and reference data. The recent advances in artificial intelligence has made it feasible to obtain great performance in finding patterns in data and extract informative high-level features (Emmert-Streib *et al.*, 2020). An increasing amount of researches have shown that deep learning technologies, such as word2vec (Zeng *et al.*, 2018), convolutional neural networks (Chen *et al.*, 2021b; Liu *et al.*, 2018), long short-term memory networks (Li *et al.*, 2019), generative adversarial networks (Liu *et al.*, 2019) and deep generative neural network (Liu *et al.*, 2021), perform exceptionally well in bioinformatics research. In the area of scRNA-seq, there is also a few deep learning approaches developed by pioneers. For example, ACTINN (Ma and Pellegrini, 2020) uses a multi-layer perceptron (MLP) to extract the high-level features of scRNA-seq data and then automatically identifies cell types. Besides, graph representation learning also wildly applies to single-cell biology (Hetzel *et al.*, 2021). scGNN is a graph neural network aggregating cell–cell relationships for gene imputation and cell clustering (Wang *et al.*, 2021). scFEA is a graph neural network leveraging the metabolic network structure to infer the cell-wise fluxome from scRNA-seq data (Alghamdi *et al.*, 2021). Though many cell identification methods have been proposed, no method is robust enough when applied to data generated from different pipelines (Abdelaal *et al.*, 2019). Besides, these aforementioned methods regard gene expression as the input feature and rarely take the relationships among genes into consideration.

However, research has shown that the gene interactions implicated in gene regulatory network or protein–protein interaction (PPI) network are informative in the different biological contexts. For instance, GCNN (Bigness *et al.*, 2022) integrates long-range regulatory interactions from Hi-C map to predict gene expression. DCell (Ma *et al.*, 2018) is a visible neural network leveraging large complexes signaling pathways to interpretably predict cell growth with gene-disruption genotypes as model input. Moreover, previous research had revealed that the joint analysis of scRNA-seq data with prior gene interaction information can lead to a meaningful understanding of data. NetNMF-sc (Elyanow *et al.*, 2020) is a network-regularized non-negative matrix factorization designed for scRNA-seq analysis, which takes advantage of prior gene network to get a more meaningful low-dimensional representation of genes. Inversely, scRNA-seq data also contain abundant information to infer gene–gene interactions (Fiers *et al.*, 2018).

Motivated by the above understandings, we propose scGraph, a graph neural network-based computational approach that takes advantage of the gene interaction network to overcome technical noise and automatically identify cell types. By integrating gene expression and gene interaction information, scGraph can not only be used to identify the cell type of individual cells, but also learn crucial gene interaction relationships from experimental data. By benchmarking scGraph against eight state-of-the-art methods on eight datasets across different species, i.e. *Homo sapiens* and *Mus musculus*, the results reveal that scGraph consistently outperforms all of the baseline methods. At last, we trained scGraph on Human Cell Landscape (HCL) dataset (Han *et al.*, 2020) and directly identified cell types of another human scRNA-seq dataset with the trained model, which demonstrated the ability of scGraph to accurately identify cell types with reference dataset.

2 Materials and methods

2.1 scRNA-seq datasets

We collected eight publicly available datasets for benchmarking our method. The T cells in colorectal cancer are generated from SmartSeq2 protocol, which can observe about 23 459 genes and 640 000 read counts per cell (Zhang *et al.*, 2018). In addition, we collected two peripheral blood mononuclear cells (PBMCs) dataset (Kang *et al.*, 2018; Zheng *et al.*, 2017), both of which are sequencing using 10x protocol. Besides, three human tissue datasets were gathered including one pancreas dataset from Baron *et al.* (2016) and two lung datasets (Lambrechts *et al.*, 2018; Travaglini *et al.*, 2020). We noted that Travaglini *et al.* built a lung cell atlas on normal lung tissue with SmartSeq2, which allowed them to annotate detailed cell types and Lambrechts *et al.* sequenced lung cancer tissues with 10x protocol, in which cells are annotated at the level of major cell type. To benchmark the performance on a large dataset with numerous batches and complex cell types, we collected a human cell atlas dataset from HCL Project (available at https://figshare.com/articles/HCL_DGE_Data/7235471) and two mouse datasets, including a mouse visual cortex dataset [Allen Mouse Brain (AMB) dataset] from Tasic *et al.* (2018) and a mouse cell atlas dataset, i.e. Tabula Muris (TM) from Tabula Muris *et al.* (2018). Since Abdelaal *et al.* also used Zheng's PBMCs dataset, Baron's pancreas dataset, AMB and TM dataset for benchmarking, these datasets can be directly downloaded from Zenodo (https://doi.org/10.5281/zenodo.3357167). We summarized the information of datasets in Supplementary Table S1.

In the data preprocessing, we first filtered out cell types/subtypes with fewer than 10 cells, unclear annotations or annotated as outliers. Then, expression data of each cell is normalized by dividing it by its total expression value and multiplying by a scale factor 10^6. We assume that the gene expression read counts follow the negative binomial distribution. Therefore, we added a pseudo count and then applied log2 transformation for each scaled expression value. The pseudo count is added to avoid any invalid logarithmic transformation when the raw read count value is zero.

2.2 Gene interaction networks

scGraph leverages gene interaction relationship for aggregating neighbor information for each gene and thus improving cell embedding and cell identification. We collected seven different human gene interaction networks and one mouse gene interaction network to evaluate the performance of scGraph as different backbone networks. One of the most well-known gene interaction networks is the STRING database (Szklarczyk *et al.*, 2015), a PPI network, which collects and integrates protein–protein association information from multiple resources, such as literatures and experiments. HumanNet (Hwang *et al.*, 2019), is a human functional gene network, which integrates diverse types of omics data by Bayesian statistics framework. The HumanNet comprises a hierarchy of human gene networks, i.e. human-derived PPIs, co-functional links, co-citations and interologs from other species. Specifically, we used two versions of HumanNet, HumanNet-CF and HumanNet-PI, which comprise of co-functional networks and PPI networks, respectively. FunCoup (Ogris *et al.*, 2018) are genome-wide functional association networks using a unique redundancy weighted Bayesian integration to combine functional association data of 10 different types. GeneMANIA (Franz *et al.*, 2018) creates a combined gene network by weighting multiplex functional genomic datasets. Besides, we collected two functional similarity matrices from pgWalk (Jiang, 2015), which are derived from KEGG pathway and Gene Ontology biological process individually. We next converted these two similarity matrices to gene networks by filtering out those gene pairs with similarity values less than a certain threshold (i.e. 0.9). These two networks are termed as pgWalk-kegg and pgWalk-gobp separately. The detailed information of gene interaction networks used in this study is summarized in Supplementary Table S2.

We noted that when applying a gene interaction network to a certain dataset, only those interaction pairs for which both

interacting genes appear in this dataset were retained and the rest of the pairs were discarded. In other words, the number of interaction pairs of a gene interaction network for different datasets could vary from each other. To capture the two regulatory directions in a pair of genes and their corresponding strengths, the gene interaction network is considered to be a directed graph, so for an edge of A gene and B gene from an undirected gene network, such as STRING PPI network, we considered it as a pair of edges (i.e. the edge from A to B and the edge from B to A). We additionally added pseudo self-interaction pair of each gene to the gene interaction network in order to aggregate information from their neighboring genes while retaining information about the genes themselves.

2.3 Structure of scGraph

scGraph is a graph neural network, taking scRNA-seq data and gene interaction network as model inputs to automatically predict the cell label. scGraph, as illustrated in Figure 1, consists of three modules: (i) a graph representation module, (ii) a feature extraction module and (iii) a classification module. The interaction relationship among genes can be presented in graph format spontaneously where a graph neural network is applied for modeling such kind of relationship. In the graph convolutional layer, whereas every node represents a gene, the edge between two nodes represents the relationship of these two corresponding genes. The graph representation module, designed as one graph convolutional layer, updates each node by aggregating the information of its neighbor nodes. We use a modified GraphSAGE convolutional layer (Hamilton *et al.*, 2017) in the graph representation module. The original update formula of GraphSAGE can be represented as

$$h_v^{k+1} \leftarrow \sigma\left(W \cdot \mathrm{MEAN}\left(\left\{h_v^k\right\} \cup \left\{h_u^k, \forall u \in N(v)\right\}\right)\right),$$

where h_v^k represents the k-th layer feature vector of node v, $N(v)$ represents the neighbor nodes of node v, W represents the trainable parameters and $\sigma(\cdot)$ is the non-linear activation function.

Since in the gene network, some hub genes, such as transcription factors, are much more important than other genes. The importance of the interaction relationship could vary a lot from each other. Toward this, we designed a trainable parameter for each edge and the formula can be represented as

$$h_v^k \leftarrow \sigma(W \cdot \mathrm{MEAN}(H_v^{k-1}\delta(S_v))),$$

where H_v^{k-1} is feature matrix constructed by stacking the feature vector set $\left\{h_v^{k-1}\right\} \cup \left\{h_u^{k-1}, \forall u \in N(v)\right\}$ where each column represents for a feature vector in the set, S_v is the edge importance score vector for the graph convolutional layer and δ is a sigmoid activation function to ensure that the edge importance scores of different edges are scaled and comparable with each other. Here, each gene is embedded as an 8D feature after graph convolutional operation.

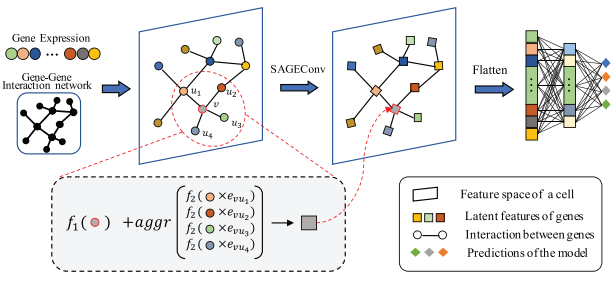

Fig. 1. The schematic overview of scGraph. Expression data are aggregated by gene network via GraphSAGE graph convolutional layer (shown in gray block) and the latent features of genes are flattened and go through the feature extraction module, i.e. two linear layers of which the high-level feature output is used to make the prediction. f is a sub-neural network, aggr indicates aggregation function, such as sum or mean functions. e_{vv} indicates the learnable edge weight of autologous interaction pair and e_{vu_1} indicates the learnable edge weight between node v and its neighbor u_1, and others are similar

In the feature extraction module, the aggregated gene features of each gene firstly go through two linear layers with 12 and 4 hidden nodes separately, and then are flatten and fed to a simple MLP of two hidden layers with 256 and 64 nodes. We use the rectified linear unit function and normalization layer after each fully connected layer. The feature extraction module reduces the dimension of aggregated gene features and the output of the module is not only used as the input of the classifier module but also used for t-SNE visualization. Finally, the classification module makes the prediction based on the high-level features extracted by the feature extraction module using a softmax function.

2.4 Model training

The parameters of scGraph are initialized with Kaiming initializer (He *et al.*, 2015). The cross-entropy loss is used for training, which can be defined as

$$L_{CE} = -\frac{1}{N}\sum_{n=1}^{N}\sum_{m=1}^{M}\left[y_n^m \cdot \log x_n^m + \left(1 - y_n^m\right) \cdot \log\left(1 - x_n^m\right)\right],$$

where N and M are the total samples and total cell types separately, y_n^m is 1 if the n-th sample belongs to m-th cell types otherwise y_n^m is 0. And x_n^m is the predicted probability of the n-th sample belonging to m-th cell type. L1 penalization of edge importance score vector S is also added to the final loss function with a regularization rate λ, which is 0.1. And the final loss function is $L = L_{CE} + \lambda \parallel S \parallel$.

We use Adam optimizer with initial settings of a learning rate equal to 0.01, and a weight decay of 10^{-4}. Cosine annealing with warm restarts learning rate strategy is firstly used to initialize model weights. Then, a learning rate strategy of reducing learning rate by a factor of 0.1 when the $F1$ metric has stopped improving is used to training the model.

Since the scRNA-seq datasets typically are unbalanced, two strategies are used to reduce the impact of an unbalanced training set. Firstly, weighted cross-entropy is used to assign different loss values for the different classes for back-propagation. Then, to avoid enormous cross-entropy weights from tiny cell types, data augmentation (see Supplementary Methods) is performed on small classes to reduce the unbalanced odds.

3 Results

3.1 scGraph outperforms baselines in automated cell-type classification task

Firstly, we benchmarked scGraph with eight baseline methods (see Supplementary Methods). We evaluated these models on eight datasets with 5-fold cross-validation in terms of mean-$F1$ (Table 1 and Supplementary Fig. S1A) and accuracy (Supplementary Table S3 and Supplementary Fig. S1B). As shown in Figure 2A, scGraph consistently outperforms all eight baseline methods on all of eight datasets. Among the eight baseline methods, ACINN and SVM are the most robust and accurate baseline methods, which are consistent with the previous study (Abdelaal *et al.*, 2019). Noticeably, scGraph is superior to ACTINN and SVM with an average 4.81% and 3.41% improvement in terms of mean $F1$ and with significant P-values of 3.90×10^{-3} on the one-side paired Wilcoxon signed-rank test for both. Specifically, in Zheng's PBMCs dataset, the mean $F1$ of scGraph is 0.877, while those for the best three baselines (SVM, ACTINN and SingleR) are and 0.853, 0.843 and 0.767, respectively. It is worth noting that the performance improvement of scGraph is more significant on complex cell identification datasets, such as Zhang's T cell dataset, which contains 20 T cells subtypes. In this dataset, although scGraph takes more time for training, scGraph outperforms SVM and ACTINN with 3.4% and 9.8% improvement in terms of mean $F1$, respectively. Furthermore, we benchmarked scGraph with SVM and ACTINN on a series of curated dataset with different number of features, i.e. genes (see Supplementary Methods). As shown in Figure 2B, scGraph outperforms SVM and ACTINN in every dataset curated with different number of highly

Table 1. Benchmark results on eight different scRNA-seq datasets in terms of mean-*F*1

	Zhang's T cells	Kang's PBMCs	Zheng's PBMCs	Lambrechts's Lung	Travaglini's Lung	Baron's Pancreas	AMB	TM
LDA	0.757	0.633	0.556	0.478	0.838	0.94	0.858	0.873
NMC	0.722	0.753	0.527	0.369	0.809	0.836	0.949	0.745
RF	0.562	0.727	0.495	0.384	0.648	0.788	0.906	0.803
SVM	0.805	0.853	0.558	0.534	0.853	0.967	0.967	0.910
SingleR	0.746	0.767	0.517	0.268	0.794	0.953	0.920	0.809
CHETAH	0.695	0.677	0.338	0.322	0.816	0.927	0.934	0.789
scID	0.508	0.692	0.498	0.342	0.589	0.463	0.782	0.563
ACTINN	0.741	0.843	0.623	0.547	0.826	0.904	0.965	0.886
scGraph	**0.839**	**0.877**	**0.681**	**0.596**	**0.861**	**0.969**	**0.976**	**0.921**

Note: The best results for each dataset are shown in **bold**.

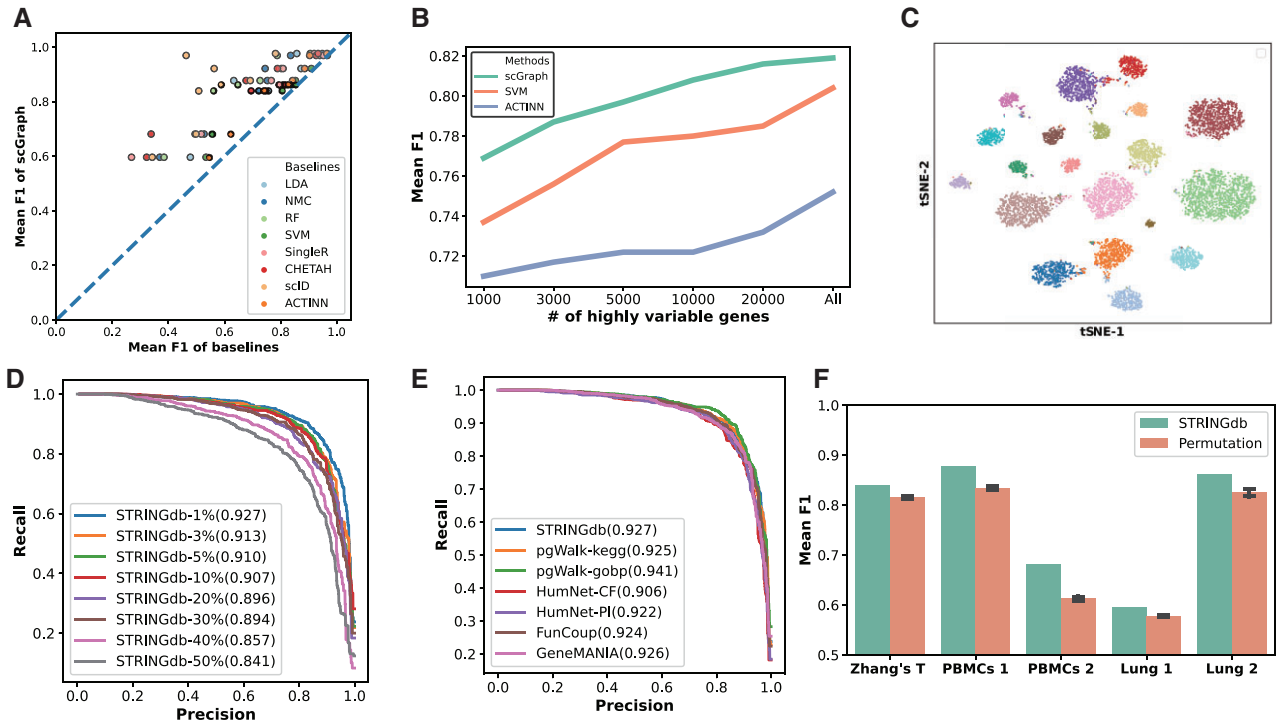

Fig. 2. Performance benchmarking. (**A**) Performance comparison of eight baselines on eight scRNA-seq datasets. The *x*-coordinate and *y*-coordinate denote the performance of a certain baseline and scGraph in terms of mean *F*1 in a same dataset. Points are colored by their corresponding baseline methods. A diagonal line is plotted in dash style. All the points are on the left top side of the diagonal line, indicating scGraph outperforms all baseline methods on all datasets. (**B**) Performance comparison of scGraph with the best two baseline methods, i.e. SVM and ACTINN on a series of Zhang's T cells datasets curated by different highly variable genes. (**C**) T-SNE plot illustrates that the cell embeddings generated by scGraph are clustered by ground-truth cell types on the Zhang's T cells dataset. (**D**) The precision–recall curves of scGraphs with STRINGdb backbone networks with different thresholds. The numbers in the parentheses represent the auPRC scores. (**E**) is similar to (D) but scGraph with different backbone networks. (**F**) Bar plot illustrates the performance of scGraph with STRINGdb backbone network and permutated STRINGdb backbone network in terms of mean *F*1 on five human datasets. (E) and (F) illustrate that scGraph achieves robust performance as long as the backbone networks are real and convincing. 'PBMCs 1', 'PBMCs 2', 'Lung 1' and 'Lung 2' indicate Kang's PBMCs, Zheng's PBMCs, Lambrechts's Lung and Travaglini's Lung dataset separately

variable genes, indicating the robust performance of scGraph to handle dataset with different scale of genes.

We also benchmarked scGraph with baseline methods on two mouse datasets, i.e. AMB and TM. To this end, the mouse PPI network from STRING database is used as the backbone network for scGraph. And scGraph still achieves the best performance against eight baseline methods, showing that scGraph can be used in different species with specie-specific backbone networks.

Technical noise is inherently contained in the data regardless of scRNA-seq pipeline, which should be removed before downstream analysis (Hwang *et al.*, 2018). To test the ability of scGraph to address the technical batch effect issue, we trained scGraph on a fetal brain dataset from the HCL project, which contains four experiment batches and six cell types. We took the high-level features of each cell from feature extraction module of scGraph and used t-SNE algorithm for dimension reduction and visualization. As illustrated in Supplementary Figure S2A and B, the cells were clustered together in the t-SNE visualization by cell types instead of by batches, indicating that scGraph can greatly overcome the batch effect and can be effectively utilized in batch effect removal.

To evaluate whether the result of scGraph is consistent with biological discoveries, we took Zhang's T cell dataset as an illustration (Supplementary Fig. S3). We firstly extracted high-level features for each cell using scGraph and applied t-SNE for dimension reduction and visualization. As shown in Figure 2C and Supplementary Figure S3A, scGraph accurately recognized the cell types for almost all the cells. Furthermore, we check the expression of biological markers for each type. Taking the biomarker IL10 as an example (Supplementary Fig. S3B), we discovered that the IL10 gene is highly and specifically expressed in the T cell subtype with IL10 as its

biomarkers. The analysis of other biomarkers, such as CD160, CX3CR1 and CXCL13, has similar results (Supplementary Fig. S3C–F). The analysis above indicates the great flexibility and performance of scGraph in cell-type identification with scRNA-seq data.

3.2 scGraph performs robustly across different gene interaction networks

We firstly checked the effect of different thresholds for STRING PPI network. We filtered the STRING network using eight thresholds so that the top 1%, 3%, 5%, 10%, 20%, 30%, 40% and 50% of the interaction pairs with the highest combined scores were retained, respectively. Then, we evaluated the performance of the scGraph with these eight STRING backbone networks on the Zhang's T cell dataset. As shown in Figure 2D and Supplementary Table S4, scGraph archives comparable performance in terms of both mean *F*1 and auPRC for the top 1–10% STRING PPI networks. But for the thresholds >10%, we observed the mean *F*1 score decreases significantly, which can be attributed to the existence of too many unconvinced interaction pairs with low combined scores in the STRING PPI network. Next, we evaluated scGraph with top 1%, 3% 5% and 10% of STRING network on six human datasets to determine the best threshold. As shown in Supplementary Tables S5 and S6, the performance of scGraph using these different STRING backbone networks is comparable in terms of mean *F*1 and auPRC. The standard deviation of mean *F*1 score of these four networks across six human datasets is 1.23%, indicating that the performance of the scGraph with these networks with different thresholds is robust. Since STRING database is widely used and the top 1% STRING network is the most convincing and condensed, we used the top 1% of STRING network as the default backbone network.

Next, we evaluated the performance of scGraph on different gene interaction networks as backbone network used in scGraph. We collected four other human gene interaction networks, i.e. HumanNet-CF, HumanNet-PI, GENMANIA and FunCoup, from three databases. Note that we only kept the top 1% of highest scoring interaction pairs for GENMANIA and FunCoup to construct gene interaction networks respectively, as their large interaction pairs which contain plenty of ambiguous edges. We also collected two crafted functional networks from pgWalk, which are built based on the functional similarity (See Materials and Methods). We compared the performance of scGraph with different backbone networks on the six human datasets. As shown in Supplementary Tables S7 and S8, scGraph achieves comparable performance with different backbone networks. The standard deviation of scGraph with different backbone networks across six human datasets is 0.013. And Figure 2E illustrates the performance of scGraph in different backbones straightforwardly in terms of precision–recall curve on Zhang's T cells dataset. From these results, we concluded that scGraph is robust with different backbone networks, which can be derived from various gene interaction databases.

For contrast, we additionally evaluated the performance of scGraph with random backbone network to verify the effectiveness of gene interaction network. We randomly shuffled the backbone networks for 10 times and evaluated on different datasets. As shown in Figure 2F, the performance of scGraph with random backbone networks in terms of mean *F*1 decreases significantly with 3.78% in average compared with those with corresponding backbone network. The analysis above indicates scGraph is very robust over different backbone networks as long as the backbone network implicates valid gene–gene interaction information even these networks differ not only in the number of nodes and edges in the networks, but also in the network function types.

3.3 scGraph accurately and adaptively identifies cell types

We firstly demonstrated that the generalization ability of scGraph to overcome not only the technical noise but also the designed perturbation. scRNA-seq experiments are usually conducted with notable differences in capturing time, equipment and even technology platform, which could possibly introduce technical noise to the data. To analyze the technical noise caused by different laboratories, we collect two human pancreas scRNA-seq datasets using CEL-seq2 and SmartSeq2 protocols from different laboratories separately and conducted a similar analysis as above. As illustrated in Supplementary Figure S4A and D, the cells were well clustered by cell types instead of laboratory categories, indicating that scGraph can overcome technical noise introduced by technicians. Furthermore, scGraph can also accurately predict cell types regardless of designed perturbation, which is an essential advantage for the cell-type classifier to be widely used in different scenarios, such as *in vivo*, *in vitro* and other stimulating conditions. To this end, we firstly collected Kang's PBMCs dataset, in which there are experimental groups of PBMCs following exposure to the cytokine interferon beta (IFN-β) along with control groups of normal PBMCs. The experimental groups of PBMCs are exposed to the cytokine IFN-β. These two groups of cells were expressed in significantly different patterns and they can also be easily separated by their experimental conditions in the t-SNE plot (Supplementary Fig. S4B and E), which is generated by the general scRNA-seq unsupervised processing analysis in the original paper. We trained the scGraph on control group of the dataset and directly make cell-type prediction on the treatment group. We collected cell embeddings and predicted cell types by similar analysis as above. As illustrated in Supplementary Figure S4C and F, scGraph is able to overcome the variations in different experimental conditions and accurately predict the cell type for experimental groups. Overall, scGraph can not only well address the technical noise introduced by different scRNA-seq protocols and different laboratories, but also overcome variants induced by designed perturbation.

To verify whether scGraph can accurately identify cell types with pretrained model, we collected three human pancreas datasets. We trained the scGraph model in Baron's pancreas dataset since this dataset has a large library size and directly identified cell types for the other two pancreas datasets. As shown in Figure 3, scGraph accurately recognizes cell types for most of the cells both in Muraro's dataset (Fig. 3A) and Segerstolpe's dataset (Fig. 3B). For instance, scGraph precisely recovered 97.3%, 95.3%, 90.2%, 98.4% and 98.1% of alpha cells, beta cells, ductal cells, delta cells and gamma cells for Muraro's pancreas dataset, respectively.

To evaluate the ability of scGraph for processing a large scRNA-seq dataset, we construct a whole human cell-type automated classification model, training on the whole HCL dataset, which including 59 human tissues and 63 cell types. We firstly trained scGraph on this reference dataset and the confusion matrix demonstrates the high accuracy of the scGraph model (Supplementary Fig. S5A). Next, we validated the performance of scGraph on other standalone human scRNA-seq datasets, assuming that they are the new sequenced scRNA-seq datasets. If the predicted probabilities of all cell type for a certain cell are smaller than a threshold, i.e. 0.1 in here, scGraph would reject to make a classification. In other words, this cell may belong to a new cell type that is not in the reference dataset. We made cell-type predictions on Kang's PBMC dataset by the trained scGraph model. As shown in Supplementary Figure S5B, almost all the cell types in the PBMC dataset are projected to correct references. The analysis above demonstrates the utility of scGraph in cell-type automated identification.

3.4 scGraph reveals important gene interaction relationship

It is worth noting that scGraph not only achieves the state-of-the-art performance but also learns the gene interaction relationship from the edge importance score vector S. Notice that the edge importance score vector S is updated along the model training process, once the scGraph model finishes training, the weights of gene–gene international network are obtained. Then, we sorted the gene–gene interaction pair based on their edge importance score s and selected the top unique target genes for downstream analysis.

To demonstrate that scGraph can learn consistent essential genes, we collected five lists of top 50 target genes from five trained

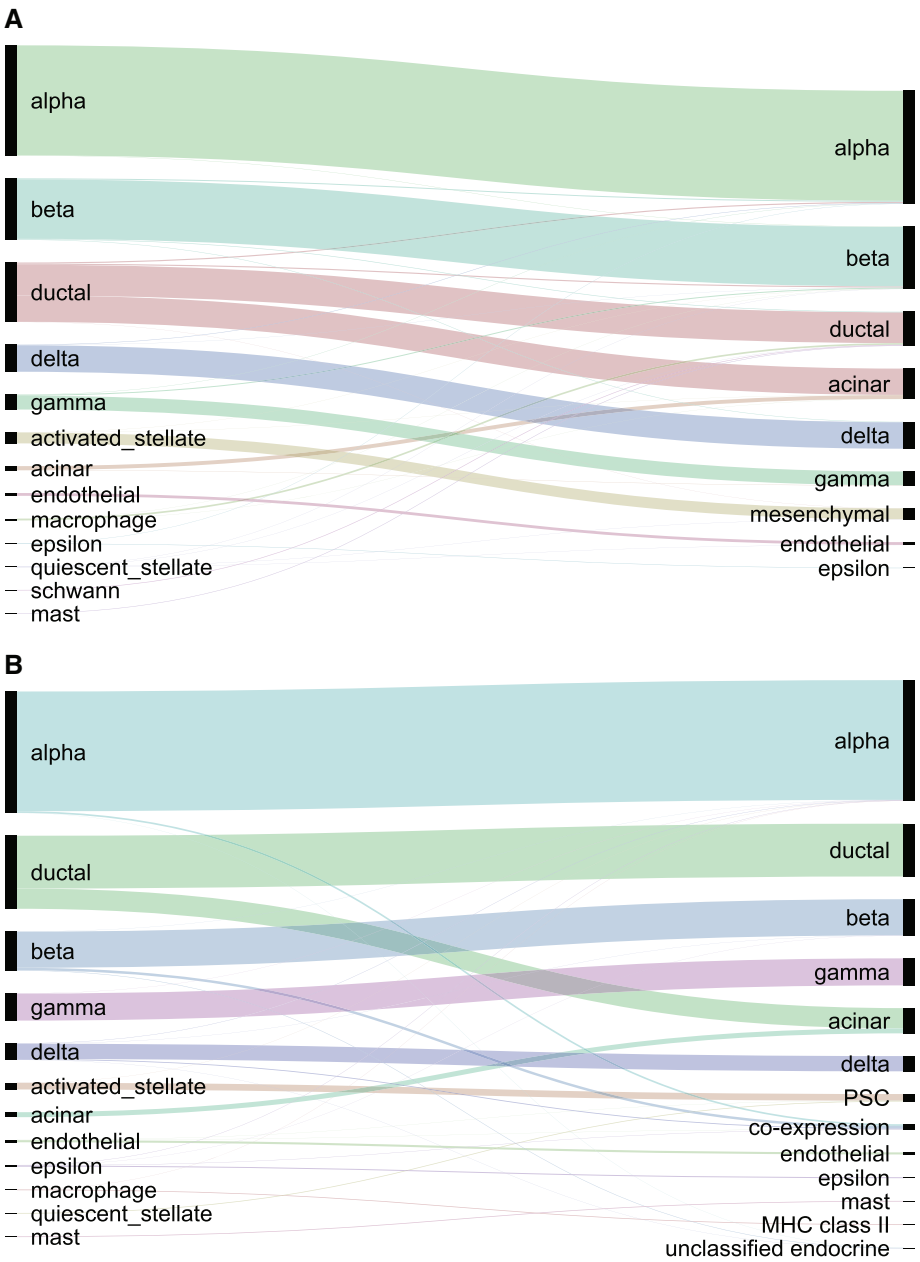

Fig. 3. Cell-type identification for newly sequencing cells. Sankey diagrams illustrate the cross-dataset validation performance on Muraro's pancreas dataset (**A**) and Segerstolpe's pancreas dataset (**B**). The breadth of each curve indicates the number of cells

scGraph model with STRING backbone network, which are trained from different cross-validation folds of Zhang's T cell dataset. As shown in Figure 4A, these five lists of essential genes are consistent with each other with averaged overlap odd (see Supplementary Methods) of 79.2% and consist of 93 unique genes, which is termed as the combined essential gene list. Similar results can be conducted for the scGraph model with HumNet-PI and pgwalk-KEGG backbone networks separately (Supplementary Fig. S6A and B). Next, we demonstrated that the essential gene lists prioritized by different backbone networks are consistent with each other. As shown in Figure 4C, the combined gene lists for STRING, HumNet-PI and pgwalk-KEGG backbone networks consist of 93, 96 and 97 genes separately (see Supplementary Table S9). The averaged overlap odd is 49.7%, while this percentage for background is 0.003% (see Supplementary Methods), indicating the consistency of essential gene list prioritized by different backbone networks. This result also explains why scGraph archived similar performance on the different backbone networks. The analysis above demonstrated that scGraph can discover consistent essential genes robustly from different backbone networks.

Next, we demonstrated the combined essential genes discovered by scGraph aggerates information from both dataset and backbone network. As shown in Supplementary Table S10, there are only 6 and 13 genes overlapped with top 100 high variable genes and top 100 high expressed genes for the 93 combined essential genes derived from scGraph with STRING backbone network. Similar analysis was carried out on scGraph with HumNet-PI and pgwalk-KEGG backbone networks. These results further support the conclusion that the discovery of the essential gene should not only depend on the expression level or variation level of gene but also the position of gene in the gene interaction networks.

Furthermore, to demonstrate that the combined essential gene list derived by scGraph is tissue-specific, we first conducted a combined essential gene list of 101 essential genes on Baron's pancreas

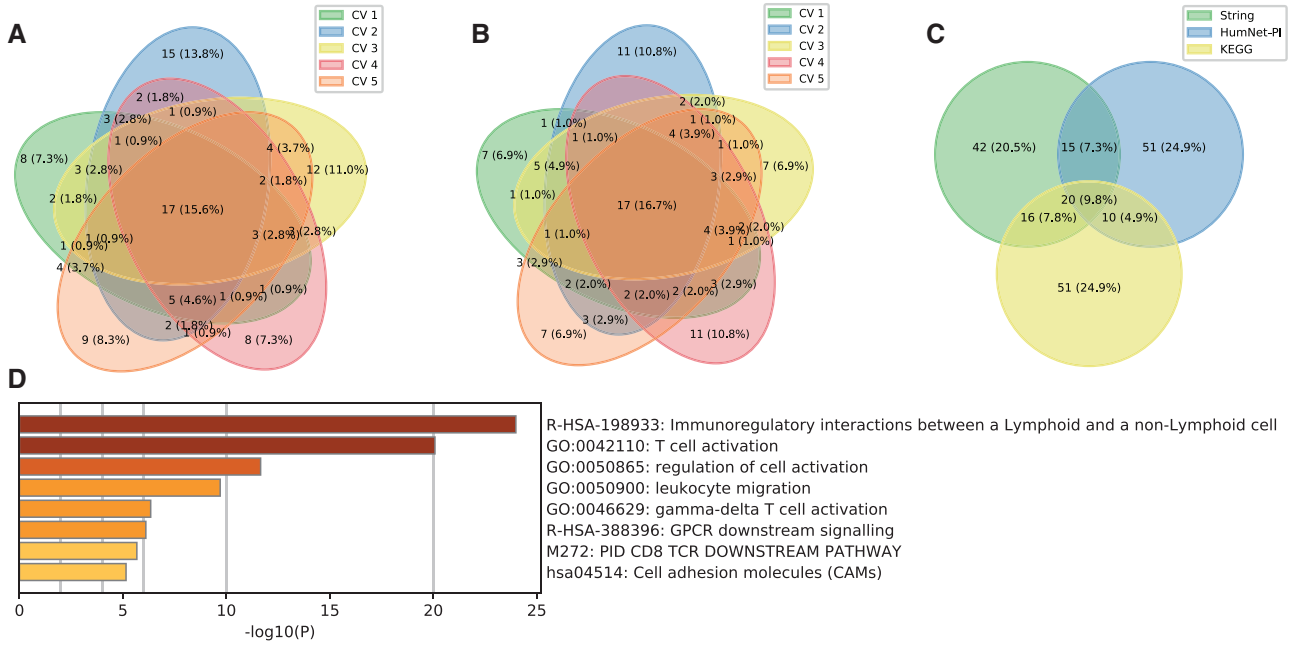

Fig. 4. Essential genes and gene pathways recovered by scGraph. (**A**) Venn plot of five essential gene lists extracted from five scGraphs with STRINGdb backbone network trained in 5-fold cross-validation manner on the Zhang's T cells dataset. (**B**) is similar to (A) but on the Baron's pancreas dataset. (**C**) Venn plot of three combined essential gene lists extracted from scGraphs with STRINGdb, HumNet-PI and pgwalk-KEGG backbone networks, respectively. (**D**) Pathway analysis recovers essential and highly related pathways based on the edge importance score vector S for the Zhang's T cells dataset

dataset by similar analysis (Fig. 4B). Interestingly, there was zero overlap between two curated gene lists from T cells dataset and pancreas dataset. Then, we conducted pathway enrichment analysis by MetaScape (Zhou *et al.*, 2019). The pathway enrichment results of T cell and pancreas datasets are illustrated in Figure 4D and Supplementary Figure S6C, respectively. For Zhang's T cells dataset, most of the pathways are highly related to the immune response. The most significantly enriched GO biology process is GO: 0042110 (T cell activation) with the multi-test adjusted *P*-value of 9.55×10^{-17}, which describes the change in morphology and behavior of a mature or immature T cell resulting from stimulus. Reactome gene sets enrichment analysis reveals that the most significant pathway is R-HSA-198933 (immunoregulatory interactions between a lymphoid and a non-lymphoid cell) with the multi-test adjusted *P*-value of 2.34×10^{-20}, which consists of a number of receptors and cell adhesion molecules play an essential role in modifying the response of immune cells to self, pathogenic organisms and tumor antigens, as a part of adaptive immune system. It convincingly demonstrates the functions of these T cells, which are sampled from colorectal tumors and adjacent normal tissues. As for the pancreas dataset, scGraph also discovers essential pancreas-specific pathways including Reactome pathway R-HSA-420092 (glucagon-type ligand receptors) and GO Biological Processes GO: 0033762 (response to glucagon). Pathway enrichment analysis illustrated above sufficiently sketch out the common attribution for corresponding datasets, indicating that scGraph efficiently leverages the gene interaction backbone network and accurately learns the tissue-specific gene–gene relationships from scRNA-seq data.

4 Discussion

We propose scGraph, a computational framework consisting of a graph neural network for automatical cell identification. We firstly benchmarked scGraph against eight baseline methods, including SVM and ACTINN, on eight datasets. The results showed that scGraph can distinguish cell types and subtypes accurately, revealing its superior performance over comparison methods. After demonstrating that scGraph is robust from different gene backbone networks, we then designed a series of experiments on different condition datasets and illustrated the performance of scGraph, in terms of visualization, robustness, scalability and flexibility.

To further illustrate this advantage of scGraph, we leveraged the edge importance score vector of the scGraph model trained on certain dataset for discovering the tissue-specific essential genes. Through a series of experiments, we found that the essential genes discovered by scGraph are reasonable and consistent with multiple runs. The pathway enrichment on these essential genes also reveals that the scGraph is capable of extracting the meaningful tissue-specific gene–gene interaction information from different datasets.

In this study, we illustrated that graph neural network is powerful to extract meaningful features and provide biological insights based on the scRNA-seq profile and the backbone network, thus shedding light on the understanding of gene regulatory mechanism. Certainly, there are some aspects of our work for improvement. First, the performance of scGraph to identify rare cell types needs to be improved, which is important for many biological processes. Second, the training procedure of scGraph needs to improve to reduce computational time. Then, it is worthwhile to embed the pathway information or the GO ontology biological processes information into the model, just similar to DCell, which embeds the biological ontologies into the model to predict the growth phenotype and genetic interaction of yeast. It is also worth trying to assemble different kinds of gene interaction networks together into one model to promote the performance. Lastly, as the quickly development of other single-cell technologies, it is worth to trying integrate different omics data by graph neural network leveraging the regulatory network to analyze single-cell data. We leave the exploration in these directions to future work.

Acknowledgements

We would like to thank Haoxiang Gao from Tsinghua University for his helpful comments. We also appreciate detailed suggestions from anonymous reviewers who significantly help us improved the early version of this manuscript.

Funding

This work was supported by the National Key Research and Development Program of China [Grant No. 2021YFF1200902]; and the National Natural

Science Foundation of China [Grant Nos U1736210, 42050101, 61873141, 61721003, 61573207, 62003178].

Conflict of Interest: none declared.

References

Abdelaal,T. *et al.* (2019) A comparison of automatic cell identification methods for single-cell RNA sequencing data. *Genome Biol.*, **20**, 194.

Alghamdi,N. *et al.* (2021) A graph neural network model to estimate cell-wise metabolic flux using single-cell RNA-seq data. *Genome Res.*, **31**, 1867–1884.

Aran,D. *et al.* (2019) Reference-based analysis of lung single-cell sequencing reveals a transitional profibrotic macrophage. *Nat. Immunol.*, **20**, 163–172.

Baron,M. *et al.* (2016) A single-cell transcriptomic map of the human and mouse pancreas reveals inter- and intra-cell population structure. *Cell Syst.*, **3**, 346–360.e344.

Bigness,J. *et al.* (2022) Integrating long-range regulatory interactions to predict gene expression using graph convolutional neural networks. *J. Comput. Biol.*, [Epub ahead of print, doi:10.1089/cmb.2021.0316, March 21, 2022].

Boufea,K. *et al.* (2020) scID uses discriminant analysis to identify transcriptionally equivalent cell types across single-cell RNA-seq data with batch effect. *iScience*, **23**, 100914.

Chen,S. *et al.* (2021a) RA3 is a reference-guided approach for epigenetic characterization of single cells. *Nat. Commun.*, **12**, 2177.

Chen,S. *et al.* (2021b) DeepCAPE: a deep convolutional neural network for the accurate prediction of enhancers. *Genomics Proteomics Bioinformatics*, **19**, 565–577.

de Kanter,J.K. *et al.* (2019) CHETAH: a selective, hierarchical cell type identification method for single-cell RNA sequencing. *Nucleic Acids Res.*, **47**, e95.

Elyanow,R. *et al.* (2020) netNMF-sc: leveraging gene-gene interactions for imputation and dimensionality reduction in single-cell expression analysis. *Genome Res.*, **30**, 195–204.

Emmert-Streib,F. *et al.* (2020) An introductory review of deep learning for prediction models with big data. *Front. Artif. Intell.*, **3**, 4.

Fiers,M. *et al.* (2018) Mapping gene regulatory networks from single-cell omics data. *Brief. Funct. Genomics*, **17**, 246–254.

Franz,M. *et al.* (2018) GeneMANIA update 2018. *Nucleic Acids Res.*, **46**, W60–W64.

Hamilton,W.L. *et al.* (2017) Inductive representation learning on large graphs. In: *Proceedings of the 31st International Conference on Neural Information Processing Systems*, Long Beach, CA, USA, pp. 1025–1035.

Han,X. *et al.* (2020) Construction of a human cell landscape at single-cell level. *Nature*, **581**, 303–309.

He,K. *et al.* (2015) Delving deep into rectifiers: surpassing human-level performance on ImageNet classification. In: *2015 IEEE International Conference on Computer Vision (ICCV)*, Santiago, Chile, pp. 1026–1034.

Hetzel,L. *et al.* (2021) Graph representation learning for single-cell biology. *Curr. Opin. Syst. Biol.*, **28**, 100347.

Hwang,B. *et al.* (2018) Single-cell RNA sequencing technologies and bioinformatics pipelines. *Exp. Mol. Med.*, **50**, 1–14.

Hwang,S. *et al.* (2019) HumanNet v2: human gene networks for disease research. *Nucleic Acids Res.*, **47**, D573–D580.

Jiang,R. (2015) Walking on multiple disease-gene networks to prioritize candidate genes. *J. Mol. Cell Biol.*, **7**, 214–230.

Kang,H.M. *et al.* (2018) Multiplexed droplet single-cell RNA-sequencing using natural genetic variation. *Nat. Biotechnol.*, **36**, 89–94.

Lambrechts,D. *et al.* (2018) Phenotype molding of stromal cells in the lung tumor microenvironment. *Nat. Med.*, **24**, 1277–1289.

Li,W. *et al.* (2019) DeepTACT: predicting 3D chromatin contacts via bootstrapping deep learning. *Nucleic Acids Res.*, **47**, e60.

Liu,Q. *et al.* (2018) Chromatin accessibility prediction via a hybrid deep convolutional neural network. *Bioinformatics*, **34**, 732–738.

Liu,Q. *et al.* (2019) hicGAN infers super resolution Hi-C data with generative adversarial networks. *Bioinformatics*, **35**, i99–i107.

Liu,Q. *et al.* (2021) Simultaneous deep generative modeling and clustering of single cell genomic data. *Nat. Mach. Intell.*, **3**, 536–544.

Ma,F. and Pellegrini,M. (2020) ACTINN: automated identification of cell types in single cell RNA sequencing. *Bioinformatics*, **36**, 533–538.

Ma,J. *et al.* (2018) Using deep learning to model the hierarchical structure and function of a cell. *Nat. Methods*, **15**, 290–298.

Macaulay,I.C. *et al.* (2017) Single-cell multiomics: multiple measurements from single cells. *Trends Genet.*, **33**, 155–168.

Mezger,A. *et al.* (2018) High-throughput chromatin accessibility profiling at single-cell resolution. *Nat. Commun.*, **9**, 3647.

Ogris,C. *et al.* (2018) FunCoup 4: new species, data, and visualization. *Nucleic Acids Res.*, **46**, D601–D607.

Pliner,H.A. *et al.* (2019) Supervised classification enables rapid annotation of cell atlases. *Nat. Methods*, **16**, 983–986.

Saelens,W. *et al.* (2019) A comparison of single-cell trajectory inference methods. *Nat. Biotechnol.*, **37**, 547–554.

Szklarczyk,D. *et al.* (2015) STRING v10: protein-protein interaction networks, integrated over the tree of life. *Nucleic Acids Res.*, **43**, D447–D452.

Tabula Muris,C. *et al.* (2018) Single-cell transcriptomics of 20 mouse organs creates a Tabula Muris. *Nature*, **562**, 367–372.

Tasic,B. *et al.* (2018) Shared and distinct transcriptomic cell types across neocortical areas. *Nature*, **563**, 72–78.

Travaglini,K.J. *et al.* (2020) A molecular cell atlas of the human lung from single-cell RNA sequencing. *Nature*, **587**, 619–625.

Wang,J. *et al.* (2021) scGNN is a novel graph neural network framework for single-cell RNA-Seq analyses. *Nat. Commun.*, **12**, 1882.

Yuan,Y. and Bar-Joseph,Z. (2019) Deep learning for inferring gene relationships from single-cell expression data. *Proc. Natl. Acad. Sci. U S A*, **116**, 27151–27158.

Zeng,W. *et al.* (2018) Prediction of enhancer-promoter interactions via natural language processing. *BMC Genomics*, **19**, 84.

Zhang,L. *et al.* (2018) Lineage tracking reveals dynamic relationships of T cells in colorectal cancer. *Nature*, **564**, 268–272.

Zheng,G.X. *et al.* (2017) Massively parallel digital transcriptional profiling of single cells. *Nat. Commun.*, **8**, 14049.

Zhou,Y. *et al.* (2019) Metascape provides a biologist-oriented resource for the analysis of systems-level datasets. *Nat. Commun.*, **10**, 1523.