Briefings in Bioinformatics, 2025, **26(3)**, bbaf198
https://doi.org/10.1093/bib/bbaf198
Problem Solving Protocol

Decoupled GNNs based on multi-view contrastive learning for scRNA-seq data clustering

Xiaoyan Yu[1], Yixuan Ren[2], Min Xia[2], Zhenqiu Shu[2,*], Liehuang Zhu[3]

[1]School of Computer Science and Technology, Beijing Institute of Technology, Zhongguancun South Street, Haidian, Beijing, 100081, China
[2]Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Jingming South Road, Chenggong, Kunming, Yunnan, 650500, China
[3]School of Cyberspace Science and Technology, Beijing Institute of Technology, Zhongguancun South Street, Haidian, Beijing, 100081, China
[*]Corresponding author. Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Chenggong, Kunming 650500, China. E-mail: shuzhenqiu@163.com

Abstract

Clustering is pivotal in deciphering cellular heterogeneity in single-cell RNA sequencing (scRNA-seq) data. However, it suffers from several challenges in handling the high dimensionality and complexity of scRNA-seq data. Especially when employing graph neural networks (GNNs) for cell clustering, the dependencies between cells expand exponentially with the number of layers. This results in high computational complexity, negatively impacting the model's training efficiency. To address these challenges, we propose a novel approach, called decoupled GNNs, based on multi-view contrastive learning (scDeGNN), for scRNA-seq data clustering. Firstly, this method constructs two adjacency matrices to generate distinct views, and trains them using decoupled GNNs to derive the initial cell feature representations. These representations are then refined through a multilayer perceptron and a contrastive learning layer, ensuring the consistency and discriminability of the learned features. Finally, the learned representations are fused and applied to the cell clustering task. Extensive experimental results on nine real scRNA-seq datasets from various organisms and tissues show that the proposed scDeGNN method significantly outperforms other state-of-the-art scRNA-seq data clustering algorithms across multiple evaluation metrics.

Keywords: clustering; scRNA-seq data; multi-view; decoupled; GNNs; contrastive learning

Introduction

Cells are the fundamental building blocks of all living organisms, playing an indispensable role in various biological functions. With the emergence of large-scale single-cell RNA sequencing (scRNA-seq) data [1, 2], developing efficient data analysis methods has become crucial for uncovering biological mechanisms. However, a clustering algorithm that is both highly accurate and clinically relevant is a significant challenge due to the complexity of biological systems [3–5].

The early scRNA-seq studies primarily relied on traditional machine learning and pattern recognition techniques [6–8], such as t-distributed stochastic neighbor embedding (t-SNE) [9], principal component analysis (PCA) [10], and nonnegative matrix factorization (NMF) [11], to extract the feature information of scRNA-seq data. Specifically, t-SNE is a commonly used nonlinear dimensionality reduction technique that reveals the relationships between cells. It transforms the similarity between cells into probabilities and minimizes the Kullback–Leibler divergence using gradient descent, effectively capturing the intrinsic structural relationships within the data. PCA is a classic dimensionality reduction method that reduces linear dependencies between samples using an orthogonal transformation. In comparison with PCA, NMF extracts latent features from the scRNA-seq data by decomposing the original data matrix into two nonnegative components. It not only maintains the nonnegativity of the data but also more effectively uncovers the sparsity and local structure within the data. Moreover, NMF learns feature representations more aligned with biological significance by enforcing nonnegativity in the representations. For example, SIMLR [12] constructs an enhanced inter-cell similarity matrix using multi-kernel learning and then obtains the final clustering results through spectral clustering. Scanpy [13] turns the cell clustering task into a community detection problem. The Leiden [14] and Louvain [15] algorithms are used for clustering after dimensionality reduction through PCA. RGNMF-DS [16] employs matrix factorization by imposing the similarity and dissimilarity regularizers while also incorporating a graph regularizer to preserve the local geometric structure of scRNA-seq data. However, although these traditional methods have played an important role in scRNA-seq data clustering in the last few decades, they fail to fully explore the complex nonlinear structure relationships in large-scale scRNA-seq data due to their limited feature representation ability.

To address the limitations of traditional methods, recent studies have developed deep clustering algorithms [17–21] specifically for scRNA-seq data analysis. Most of them attempt to seamlessly integrate dimensionality reduction and clustering into an end-to-end framework, significantly enhancing overall analysis performance. Notably, autoencoder-based scRNA-seq data clustering

Received: January 12, 2025. **Revised:** March 25, 2025. **Accepted:** April 07, 2025

methods [22, 23] have demonstrated excellent performance in real-world applications. For example, ADClust [24] first generates low-dimensional representations using an autoencoder and applies the Dip-test [25] for statistical testing to mine the differences between micro-clusters. Then similar micro-clusters are merged until convergence by jointly optimizing the clustering and autoencoder loss functions. scMCKC [26] improves cell clustering performance by incorporating multiple constraints to capture cell similarities, followed by the application of a soft k-means algorithm for clustering analysis. DeepScena [27] employs a negative binomial model-based autoencoder for more accurate data imputation. Additionally, it integrates a self-supervised deep clustering model with a top-down hierarchical clustering strategy to improve data similarity and feature precision. scINDC [28] aims to maximize the mutual information between the input and output representations of the autoencoder, thus enhancing its efficiency.

However, autoencoder-based scRNA-seq data clustering methods primarily focus on learning the feature representation of gene expression, neglecting the relationships between cells. These relationships can effectively reveal their similarities and provide meaningful insights for clustering analysis. To address this issue, graph neural networks (GNNs) have been introduced to model the relationships between nodes effectively. In recent years, several GNN variants [29–32] have been widely applied to scRNA-seq data clustering. They utilize intercellular relationships to better capture both global and local cell features, thereby significantly improving clustering performance. For example, scCDG [33] combines a denoising autoencoder based on the zero-inflated negative binomial distribution [34] with GNNs to preserve the topological structure of scRNA-seq data. scDFN [35] integrates an autoencoder and an enhanced graph autoencoder, utilizing cross-network information fusion and a triple self-supervised strategy to significantly improve single-cell data clustering performance.

Although the aforementioned scRNA-seq clustering methods achieve promising performance on limited samples, relying on a single perspective is insufficient to capture the multidimensional features and complex relationships between cells. In recent years, various multi-view clustering methods [36–38] have been developed for scRNA-seq data clustering. ScCCL [39] utilizes a self-supervised contrastive learning approach by incorporating random masking of gene expression and noise addition. It combines momentum encoding and cluster-level contrastive learning, significantly enhancing the scRNA-seq data clustering performance. scMMN [40] constructs shallow and deep views to achieve comprehensive feature representation. Then it combines group contrastive loss with structural consistency contrastive loss to optimize the model. However, as the number of network layers and views increases, the complexity of these models grows significantly in clustering tasks. This not only consumes expensive computational costs but also may lead to overfitting, thereby affecting the model's generalization ability. Therefore, improving training efficiency without compromising model effectiveness poses a huge challenge for existing multi-view scRNA-seq clustering methods.

To address the above challenges, we introduce decoupled GNNs based on multi-view contrastive learning (scDeGNN) for scRNA-seq data. Firstly, we construct two different adjacency matrices to generate two views. Then, these two views are input into the decoupled GNNs for training, generating initial cell feature representations. These representations are then passed through a multilayer perceptron (MLP) for further feature refining, generating cell feature representation of each view. To enhance the model's robustness and the consistency of the representations, we introduce a contrastive learning mechanism in our proposed method. This mechanism refines the newly generated representations to ensure both feature consistency and discriminability. Finally, these two learned representations are concatenated to form a common feature representation of multi-view scRNA-seq data, which is then used for the final clustering task. Extensive experiments show that the proposed scDeGN method achieves better performance than other state-of-the-art scRNA-seq data clustering methods on several scRNA-seq datasets.

The contributions of this work are summarized as follows:

1. We construct a multi-view scRNA-seq data learning framework based on decoupled GNNs to decompose complex learning tasks into simpler sub-tasks. Each sub-network specializes in different features or local information, enhancing more efficient learning. On the one hand, the decoupled GNNs train multiple sub-networks in parallel. This allows each sub-network to focus on different dimensions of the cell data, thereby enhancing the model's representation ability. Additionally, the proposed approach models scRNA-seq data from multiple perspectives, capturing the diversity of cell relationships more comprehensively.
2. We introduce a multi-view contrastive learning mechanism in our proposed method. It first creates two views by constructing two distinct algorithms, which are then independently trained using decoupled GNNs. By employing a contrastive learning strategy, we refine the learned representations to ensure consistency while preserving strong distinguishability. This method strengthens the model's robustness and significantly improves the cell representations, thereby enhancing the accuracy and stability of clustering results.
3. Experimental results on nine real scRNA-seq datasets demonstrate that our scDeGNN method significantly improves the stability and accuracy of clustering in high-dimensional, complex scRNA-seq data applications.

The structure of this paper is as follows: Section 2 describes the proposed scDeGN method, Section 3 presents the experimental results and analysis, and Section 4 concludes the paper.

Methods

In this section, we will provide a comprehensive introduction to the proposed scDeGNN method.

Overview

In this paper, we propose decoupled GNNs based on multi-view contrastive learning for scRNA-seq data clustering. This method mainly consists of five modules: data preprocessing, multi-view construction, decouple GNNs, contrastive learning, and feature fusion and clustering, as shown in Fig. 1. The data preprocessing module filters cells and genes to ensure high-quality scRNA-seq data for subsequent training. The multi-view construction module employs the K-nearest neighbors (KNN) algorithms [41] and the diffusion map matrix (DM) algorithms [42] to generate two distinct views, which are then input into the decoupled GNNs module to generate the initial cell representations. The contrastive learning module guides model training by reducing the distance between cells in the same cluster and increasing the distance between cells in different clusters. Finally, we fuse these learned features to generate a unified cell representation and then apply the k-means algorithm for clustering.

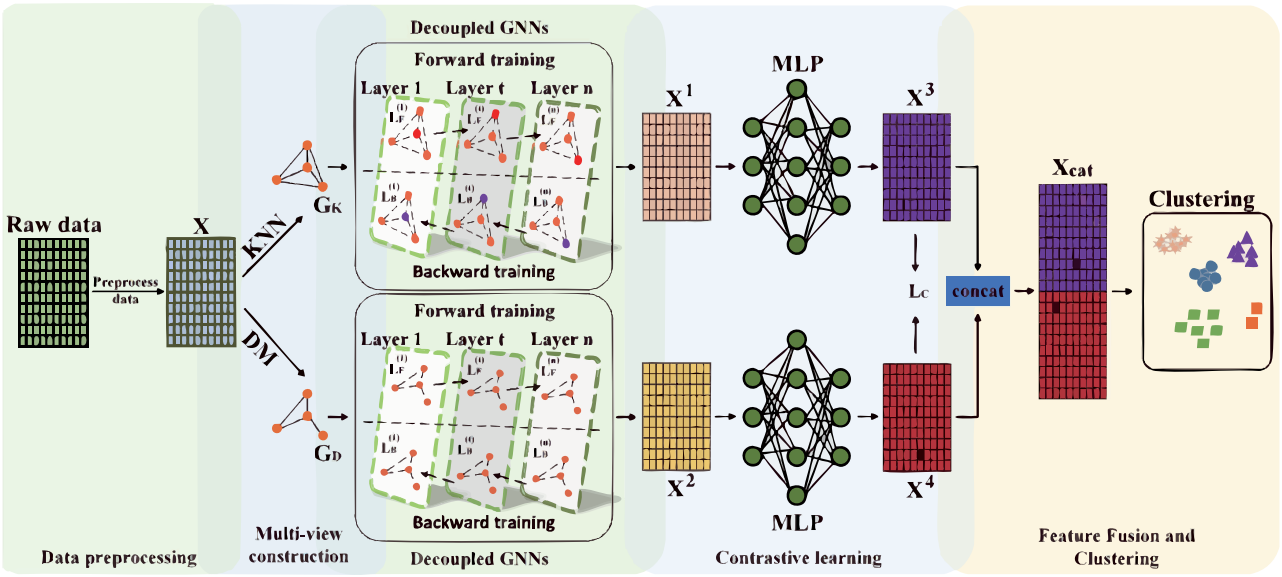

Figure 1. The overall framework of the proposed scDeGNN method.

Data preprocessing module

In this subsection, we introduce the details of cell data preprocessing. Firstly, the cell expression data are downloaded from the gene database. Non-expressed cells or genes are removed to enhance clustering stability, and the top 2000 or 3000 highly expressed genes are selected for subsequent training. Thus, we can obtain the preprocessed gene expression matrix $X \in R^{n \times m}$, where n represents the number of preprocessed cells and m is the number of genes.

Multi-view construction module

To address the limitation of single-view data, the proposed method employs a multi-view learning strategy to leverage the complementarity of multi-view data. To construct multi-view scRNA-seq data, we calculate the cell distance using the KNN and DM algorithms. Specifically, the KNN algorithm effectively captures the local data structure by identifying the nearest neighbors for each data point, making it well-suited for data with local characteristics. This enables it to preserve the detailed features of the data in each view. The DM algorithm captures the global geometric structure of the data by constructing a diffusion process. By calculating the diffusion distance between data points, it preserves global structural information at multiple levels. This is essential for visualizing and interpreting high-dimensional data. By combining these two algorithms, both linear and nonlinear structures in scRNA-seq data can be effectively exploited in our model. Specifically, we derive two graph structures, G_K and G_D, generated from the KNN and DM algorithms, respectively.

Firstly, using the KNN algorithm, the adjacency matrix A_K of the graph G_K is defined as follows:

$$A_{K_{ij}} = \begin{cases} 1, & \text{if } x_j \in \text{KNN}(x_i) \\ 0, & \text{otherwise} \end{cases}, \tag{1}$$

where KNN $(\cdot)$ denotes the set of nearest neighbors for a sample.

Next, the DM algorithm computes the nonlinear distance between cells as follows:

$$\bar{d}_{ij} = \frac{1}{2}\left(e^{\left(-\frac{d_{ij}}{2\partial_k(x_i)^2}\right)} + e^{\left(-\frac{d_{ij}}{2\partial_k(x_j)^2}\right)}\right), \tag{2}$$

where $\partial_k(x_i)$ represents the bandwidth of the Gaussian kernel function, and d_{ij} is the Euclidean distance between two cells x_i and x_j. The scaling factor is computed based on the distance distribution from the nearest neighbors, ensuring that the resulting distance reflects both local and global structures.

In the DM algorithm, the adjacency matrix A_D of the graph structure G_D is derived by setting the weight values of the first k nearest neighbor cells to 1 while setting those of the remaining cells to 0. The final adjacency matrix is symmetrized to ensure mutual relationships between cells.

Decouple GNNs module

To overcome the limitation of relying on a single complex model to process all data from different views, we employ a decoupled GNN framework to train multiple simpler GNN models in parallel. Specifically, we use the KNN and DM algorithms to generate two different views and then feed them into two separate decoupled GNNs for training. Each decoupled GNN processes the views created by KNN and DM independently, allowing each sub-model to learn the latent cell representations. This approach decouples multi-layer complex GNNs into multiple simpler modules, resulting in more efficient training.

In general, the separable GNN model can be expressed as follows:

$$f(\mathbf{A}, \mathbf{X}, \mathbf{W}) = f_1\left(f_0(\mathbf{A}, \mathbf{X}), \mathbf{W}\right), \tag{3}$$

where X represents the feature matrix, A denotes the adjacency matrix, and W is the projection coefficient. f_0 represents graph operations and f_1 represents neural operations.

Several studies [43, 44] show the GNNs are completely separable. Therefore, the separable GNNs can be expressed as follows:

$$\begin{aligned}\mathcal{F}_k = \Big\{ f : \mathbb{R}^{n\times n} \times \mathbb{R}^{n\times d} \times \mathbb{R}^{d\times k} \mapsto \mathbb{R} \mid f(\boldsymbol{A}, \boldsymbol{X}, \boldsymbol{W}) \\ = f_1\left(f_0(\boldsymbol{A}, \boldsymbol{X}), \boldsymbol{W}\right) = g_1\left(\boldsymbol{A}, g_0(\boldsymbol{X}, \boldsymbol{W})\right) \Big\},\end{aligned} \tag{4}$$

where n denotes the number of cells, d denotes the number of genes, and k denotes the number of iterations. It can be further understood from Equation (4) that the graph operation $A_s = f_0(A, X)$ is first performed to obtain a fixed graph structure, which is then input into a standard GNN layer for training. Due to the prior knowledge, when $f(\boldsymbol{A}, \boldsymbol{X}, \boldsymbol{W}) = f_1\left(f_0(\boldsymbol{A}, \boldsymbol{X}), \boldsymbol{W}\right) = g_1\left(\boldsymbol{A}, g_0(\boldsymbol{X}, \boldsymbol{W})\right)$, it can be proved that this GNNs is completely separable [44]. Therefore, the GNNs used in this paper are fully separable.

The separable GNNs consist of two parts: forward training and backward training. It is worth noting that forward training provides guidance for backward training and obtains the potential features of cells.

Forward training

Each layer of GNNs is regarded as a separate module, and the learned potential features are passed to the next module for training after each module has been trained. Therefore, each module needs to have an objective function for training.

Assume that the latent feature representation learned by the GNNs module is H, which can be defined as follows:

$$H = f(A, X, W) \tag{5}$$

In this paper, we use the GAE as the separable GNNs. Therefore, the loss function L_F for this module is defined as follows:

$$L_F = l(A, X, W) = d(A, k(H)), \tag{6}$$

where $d(\cdot, \cdot)$ denotes the metric function and $k(\cdot)$ denotes the mapping function.

Backward training

During the backward training, we aim to pass information from the current module to the previous module. Thus, the previous module can forward the effective information needed by the next module.

Here, we denote the latent features H learned by each GNN module as

$$H = f_1(f_0(A, \varphi(XU)), W), \tag{7}$$

where $U \in R^{d\times d}$ denotes learnable square matrix, and $\varphi(\cdot) : \mathbb{R}^d \mapsto \mathbb{R}^d$ denotes nonparametric function.

By integrating Equation (7) into Equation (4) and performing some series of derivations, we obtain the loss function of this module. In the separable GNNs, assuming that the latter module expects the desired feature representation obtained from the previous module to be Z, the loss function of each backward training module can be defined as follows:

$$L_B^t = d(H_t, Z_{t+1}) = d(F_t(A, X_t, U_t, W_t), Z_{t-1}), \tag{8}$$

where t denotes the tth module in the separable GNNs and F denotes the transformation function from $\mathbb{R}^{n\times n} \times \mathbb{R}^{n\times d} \times \mathbb{R}^{d\times d} \times \mathbb{R}^{d\times k}$ to $\mathbb{R}^{n\times k}$. Equation (8) aims to make the input features as similar as possible to the desired features.

Contrastive learning module

In the previous module, we obtain the representations X^1 and X^2 from the decoupled GNNs and then input them into separate MLP to generate the refined feature representations X^3 and X^4. Thus, we have

$$[X^3, X^4] = MLP(X^1, X^2) \tag{9}$$

To effectively capture the intrinsic similarities and differences between cells, we optimize the cell representations X^3 and X^4 generated by the graph structure using contrastive learning. Specifically, contrastive learning minimizes the distance between samples of the same class while maximizing the distance between samples of different classes, allowing the model to learn more discriminative features. Therefore, the contrastive learning loss function L_c is given as follows:

$$L_c = \frac{1}{2c}\sum_{i=1}^{c}\left[l\left(X_{,i}^3, X_{,i}^4\right) + l\left(X_{,i}^4, X_{,i}^3\right)\right], \tag{10}$$

where

$$l\left(X_i^a, X_i^b\right) = -\log\frac{e^{\left(\varphi\left(X_i^a, Z_i^b\right)/\tau\right)}}{\sum_{j=1,j\neq i}^{c} e^{\left(\varphi\left(X_i^a, X_j^a\right)/\tau\right)} + \sum_{j=1}^{c} e^{\left(\varphi\left(X_i^a, X_j^b\right)/\tau\right)}}, \tag{11}$$

where $\varphi(\cdot, \cdot)$ is a function used to calculate the similarity between samples. τ is a temperature hyperparameter that adjusts the sensitivity of the softmax, and c represents the total sample number.

Feature fusion and clustering module

In the final module of the proposed method, we concatenate the learned feature representations X^3 and X^4 to obtain the common representation of multi-view scRNA-seq data, which is then input into k-means for clustering. Therefore, we have

$$X_{cat} = cat\left(X^3, X^4\right), \tag{12}$$

where X_{cat} denotes the common representation of multi-view scRNA-seq data.

In summary, the loss function of our proposed method can be defined as follows:

$$L = \alpha L_F + \beta L_B + \gamma L_c \tag{13}$$

By performing this series of operations, the proposed model captures the diversity of cell relationships more comprehensively, thereby enhancing the learning efficiency and the feature representation ability.

Algorithm 1 The Algorithm of the proposed scDeGNN method

Input: Preprocessed gene expression matrix X.
Output: Cluster assignments $\{C_i\}_{i=1}^{k}$
Step 1: Multi-View Construction
Construct the adjacency matrix A_K using KNN.
Construct the adjacency matrix A_D using DM.
Step 2: Decoupled GNNs Training
for each graph structure $A \in \{A_K, A_D\}$ **do**
    Compute latent representations by Equation (5).
    Optimize forward loss by Equation (6).
    Optimize backward loss by Equation (8).
end for
Step 3: Contrastive Learning
Extract the initial feature representations X^1 and X^2 from the two GNN models.
Generate the refined feature representations X^3 and X^4 by feeding X^1 and X^2 into the MLP.
Compute contrastive loss by Equation (9).
Step 4: Feature Fusion and Clustering
Fuse the features from the two views to obtain X_{cat}.
Perform clustering on X_{cat} using k-means to obtain cluster assignments $\{C_i\}_{i=1}^{k}$. **return** Cluster assignments $\{C_i\}_{i=1}^{k}$

Experiments

In this section, we conducted several clustering experiments on nine scRNA-seq datasets to validate the effectiveness of the proposed scDeGNN method.

Evaluation metric

In our experiments, four common metrics, such as accuracy (ACC) [45], adjusted rand index (ARI) [46], normalized mutual information (NMI) [47], and Fowlkes–Mallows index (FMI) [48], were used to evaluate the performances of scRNA-seq data clustering methods. Generally, higher values for these metrics indicate better clustering results.

ACC assesses the extent to which each cluster corresponds to its true class. The ACC is calculated by

$$ACC = \frac{\sum_{i=1}^{N} 1 \text{ if } Y_i^{true} == Y_i^{pre}}{N}, \tag{14}$$

where Y_i^{true} is the true clustering label for the ith cell, and Y_i^{pre} is the predicted clustering label for the ith cell. N represents the number of cells.

NMI evaluates the closeness of clustering results to the true cluster labels. The NMI can be defined as follows:

$$NMI = \frac{\mathbf{MI}(Y^{pre}, Y^{true})}{\max(\mathbf{H}(Y^{pre}), \mathbf{H}(Y^{true}))}, \tag{15}$$

where $\mathbf{MI}(\cdot)$ denotes mutual information, and $\mathbf{H}(\cdot)$ represents information entropy.

ARI measures the similarity between the clustering label distribution and the true label distribution. The definition of ARI is given as follows:

$$ARI = \frac{RI - \mathbf{E}(RI)}{\max(RI) - \mathbf{E}(RI)}, \tag{16}$$

Table 1. The characteristics of nine real scRNA-seq datasets

Datasets	Cells	Genes	Cell types
QS_Trachea	1350	23 341	4
QS_Heart	4365	23 341	8
Trachea	11 269	23 341	5
Bladder	2500	16 867	4
Pollen	301	21 721	11
Plasschaert	6977	28 205	8
Chen	12 089	23 284	46
Romanov	2881	21 143	7
Young	5685	25 215	11

where

$$RI = \frac{TP + TN}{TP + TN + FP + FN}, \tag{17}$$

where TP and TN represent true positives and true negatives, respectively. FP and FN stand for false positives and false negatives, respectively. $\mathbf{E}(\cdot)$ is the expected value.

FMI is a measure of similarity between two clusters or sets based on the geometric mean of precision and recall. The FMI can be computed as follows:

$$FMI = \frac{a}{\sqrt{(a+b)(a+c)}}, \tag{18}$$

where a is the number of cells that belong to the same cluster in Y^{pre} and the same cluster in Y^{true}. b is the number of cells that belong to the same cluster in Y^{pre} and different clusters in Y^{true}. c is the number of cells belonging to different clusters in Y^{pre} and the same cluster in Y^{true}.

Datasets

To verify the clustering performance of the proposed model, we carried out a series of experiments on nine real scRNA-seq datasets: Quake_Smart-seq2_Trachea [49], Quake_Smart-seq2_Heart [49], Quake_10x_Trachea [49], Quake_10x_Bladder [49], Pollen [50], Plasschaert, Chen [51], Romanov [52], and Young [53]. For convenience, these datasets are abbreviated as QS_Trachea, QS_Heart, Trachea, and Bladder, respectively. Table 1 summarizes the number of cells, genes, and classes in each dataset.

Evaluation setting

To ensure fairness, all experiments were conducted using CUDA 11.6, PyTorch 1.13.1, Python 3.7, and an NVIDIA GeForce RTX 3090 GPU. For the proposed ScDeGNN method, we involved the following hyperparameters: α, β, and γ, which were set to 0.1, 0.0001, and 0.0001, respectively. Additionally, the batch size was set to 10 and the number of clusters for k-means matched the category number of each dataset. The learning rate was set to 0.0001 for the QS_Trachea, Trachea, Bladder, and Chen datasets, and 0.000001 for the remaining datasets.

Baselines

In this study, we compared the proposed method with eight already published methods. They include kmeans [54], ScCCL [39], scMCKC [26], sc-INDC [28], scCDG [33], ADClust [24], DeepScena [27], and scDFN [35]. Here, we briefly introduce these state-of-the-art scRNA-seq data clustering methods as follows:

1. k-means: this algorithm aims to cluster n samples into k groups based on their similarity, minimizing the distance between each object and the centroid of its assigned cluster. Its source code is available at https://scikit-learn.org/dev/modules/clustering.html#k-means.
2. ScCCL: this algorithm is a self-supervised contrastive learning clustering method. It constructs multiple views through data augmentation. Contrastive learning strategy operates at both the instance and cluster levels. The source code of the scMCKC method can be found at https://github.com/LuckyxiaoLin/ScCCL.
3. scMCKC: this approach learns the latent representation of cells using the autoencoders, and then significantly improves the clustering performance through an innovative compactness constraint. Its source code is available at https://github.com/leaf233/scMCKC.
4. sc-INDC: this algorithm combines an autoencoder with multiple constraints for clustering, offering the notable advantage of reduced model complexity. The source code of the scCDG method is available at https://github.com/arnabkmondal/sc-INDC.
5. scCDG: it integrates a denoising autoencoder and GCNs for training. The denoising autoencoder aims to produce denoised data, while GCNs are designed to learn the latent representations of cells. The source code of the scCDG method can be found at https://github.com/WHY-17/scCDG.
6. ADClust: the main framework of this method is the autoencoder. After getting the cell micro-clusters, it continuously trains and updates the clusters. The source code is available at https://github.com/biomed-AI/ADClust.
7. DeepScena: this method is based on the autoencoder framework. It first generates cell micro-clusters, then iteratively trains and updates these clusters to achieve the final clustering results. Its source code is accessible at https://github.com/shaoqiangzhang/DeepScena.
8. scDFN: this method utilizes an autoencoder to extract latent representations of cells and captures cell correlations using a graph autoencoder. Finally, it achieves clustering through information fusion. Its source code can be found at https://github.com/11051911/scDFN.

Comparative experiments

Tables 2–4 show the clustering performances of the proposed method and its competitors across several real scRNA-seq datasets. According to these results, we draw the following conclusions:

1. It can be seen that our method significantly outperforms other approaches in clustering tasks, achieving optimal or sub-optimal results across almost all scRNA-seq datasets. This demonstrates the effectiveness of our method in scRNA-seq data clustering. Compared with other methods, our approach has two significant advantages in clustering applications. Firstly, we develop a multi-view scRNA-seq data learning framework based on decoupled GNNs, which decomposes complex tasks into several simpler sub-tasks. This approach enables each sub-network to focus on different aspects of the cell data, thereby improving the model's representation ability. Secondly, by integrating a contrastive learning mechanism, we ensure the consistency of representations during optimization while maintaining their distinguishability, which further enhances the model's robustness and clustering performance.
2. It can be observed that all methods performed well on the Pollen dataset, but their clustering performance significantly declined on the Trachea and Chen datasets. This can be attributed to the smaller size of the Pollen dataset, which allows the model to more accurately capture the latent representations of the cells. In contrast, the Trachea and Chen datasets, with their larger number of cells and diverse cell clusters, increase the complexity of model training. This poses challenges in effectively capturing cell heterogeneity and the underlying population structure.
3. It is evident that ScCCL, sc-INDC, and our proposed method consistently perform well, often ranking first or second. This is because all three methods adopt the multi-view learning framework to learn the representations of scRNA-seq data. In contrast to single-view methods, the multi-view learning framework can fully leverage the complementarity of multi-view scRNA-seq data. These models, therefore, fully explore latent cell features from different perspectives. Furthermore, they can effectively exploit the interrelationships between multi-view scRNA-seq data, significantly improving their feature representation capacity and clustering performance.
4. The experimental results demonstrate that scDeGNN significantly surpasses scMCKC and scDFN in clustering performance across nine scRNA-seq datasets. This advantage primarily stems from its innovative integration of deep learning methods with GNNs, which breaks through the limitations of traditional multi-view strategies. Specifically, scDeGNN builds a multi-view learning framework with a decoupled GNN architecture, which effectively extracts complementary local and global features from heterogeneous adjacency matrices and merges them efficiently through a multi-view contrastive learning mechanism. In contrast to traditional multi-view methods that depend on simple feature concatenation or constraint fusion, the GNN module in scDeGNN propagates information hierarchically within graph-structured data. This approach explicitly captures the potential relationships between cells. On one hand, graph convolution operations propagate multi-view features through the cell similarity network, improving the model's ability to represent sparse, high-dimensional data. On the other hand, the contrastive learning loss function aligns the semantic spaces of different views, maintaining the distinctiveness of each view while strengthening consistency across views.

Ablation experiments

In this part, we constructed the following variants of the proposed method as follow: (1) multi-view ablation: the multi-view learning framework was replaced by a single-view framework; (2) decoupled GNNs ablation: the decoupled GNNs were replaced by a standard GNNs; (3) contrastive learning ablation: this variant removed the contrastive learning part. The above three ablations are denoted as w/o M, w/o G, and W/o C, respectively. Ablation experiments were carried out on the Pollen, Trachea, and Bladder datasets. The experimental results of the proposed method and its variants are shown in Table 5.

It can be found from the results that the proposed method outperforms other variants on these four scRNA-seq datasets. In addition, the three ablation experiments were conducted by keeping two groups unchanged while altering one group. Based

Table 2. The clustering performance comparisons on the QS_Trachea, QS_Heart, and Trachea datasets

Data	Metrics	k-means	ScCCL	scMCKC	sc-INDC	scCDG	ADClust	DeepScena	scDFN	Ours
QS_Trachea	ACC	0.5125	0.8630	0.8318	0.9278	0.4674	0.7229	0.5066	0.7148	**0.9459**
	NMI	0.5833	0.6945	0.7845	0.8210	0.4603	0.6229	0.2342	0.6583	**0.8414**
	ARI	0.3386	0.8224	0.8543	**0.8679**	0.3001	0.6187	0.1601	0.5446	0.8488
	FMI	0.5543	0.9000	0.9148	**0.9228**	0.5211	0.8140	0.5111	0.7197	0.9187
QS_Heart	ACC	0.4568	0.8937	0.9088	0.8987	0.5775	0.8224	0.6799	0.7766	**0.9601**
	NMI	0.2211	0.8562	0.8251	0.8554	0.6625	0.6164	0.7332	0.8346	**0.9007**
	ARI	0.1020	0.8952	0.8681	0.8901	0.4763	0.5805	0.6270	0.6906	**0.9309**
	FMI	0.3770	0.9289	0.9105	0.9248	0.6280	0.7614	0.7325	0.7880	**0.9536**
Trachea	ACC	0.5335	0.7276	0.6749	0.6820	0.5145	0.6468	0.4515	0.6891	**0.9472**
	NMI	0.5281	0.7371	0.6488	0.6588	0.5888	0.6575	0.4626	0.6678	**0.8247**
	ARI	0.2473	0.5447	0.5094	0.5136	0.3275	0.5484	0.2740	0.5050	**0.8960**
	FMI	0.5307	0.7423	0.7179	0.7014	0.5793	0.5617	0.5465	0.7163	**0.9517**

Table 3. The clustering performance comparisons on the Bladder, Pollen, and Plasschaert datasets

Data	Metrics	k-means	ScCCL	scMCKC	sc-INDC	scCDG	ADClust	DeepScena	scDFN	Ours
Bladder	ACC	0.7080	0.8232	0.6028	0.9953	0.5352	0.7972	0.9716	0.7744	**0.9992**
	NMI	0.5371	0.8096	0.7020	0.9700	0.6338	0.7361	0.9425	0.8095	**0.9934**
	ARI	0.5234	0.7853	0.5398	0.9871	0.4269	0.7368	0.9807	0.7609	**0.9976**
	FMI	0.7365	0.8768	0.7184	0.9929	0.6353	0.8140	0.9895	0.8620	**0.9987**
Pollen	ACC	0.8571	0.9037	0.7774	0.8946	0.7109	0.8637	0.5780	0.9136	**0.9369**
	NMI	0.9103	0.8948	0.8861	0.9299	0.7677	0.9168	0.4853	0.9270	**0.9443**
	ARI	0.8489	0.8955	0.7621	0.9133	0.6413	0.8448	0.6089	0.9352	**0.9456**
	FMI	0.8661	0.9069	0.8065	0.9240	0.6876	0.8708	0.6794	0.9423	**0.9516**
Plasschaert	ACC	0.4863	0.8482	0.7627	0.7961	0.5490	0.8780	0.7821	0.7169	**0.9470**
	NMI	0.3962	0.7715	0.6960	0.7723	0.6292	0.6384	0.7035	0.7086	**0.8183**
	ARI	0.3068	0.7734	0.6735	0.7433	0.4389	0.6572	0.6858	0.5999	**0.8708**
	FMI	0.5483	0.8578	0.7910	0.8374	0.6229	0.7992	0.7994	0.7400	**0.9208**

Table 4. The clustering performance comparisons on the Chen, Romanov, and Young datasets

Data	Metrics	k-means	ScCCL	scMCKC	sc-INDC	scCDG	ADClust	DeepScena	scDFN	Ours
Chen	ACC	0.3847	0.5923	0.6127	0.5195	0.7301	**0.7757**	0.3734	0.6049	0.7756
	NMI	0.5150	0.7201	0.7036	0.7467	0.7420	0.7871	0.5199	0.7620	**0.8115**
	ARI	0.2742	0.7250	0.5719	0.5602	0.7922	0.7929	0.3163	0.6545	**0.8097**
	FMI	0.3595	0.7665	0.6808	0.6367	0.8310	0.8088	0.4067	0.7136	**0.8387**
Romanov	ACC	0.3939	0.8474	0.6408	0.8419	0.6279	0.7698	0.5855	0.8841	**0.8980**
	NMI	0.1793	0.7189	0.5733	0.7157	0.5020	0.5517	0.5325	0.6958	**0.7412**
	ARI	0.1060	0.7600	0.4148	0.7414	0.4505	0.5911	0.4099	0.7543	**0.7783**
	FMI	0.3776	0.8178	0.5885	0.8042	0.5716	0.7159	0.5694	0.8226	**0.8350**
Young	ACC	0.4216	0.7912	0.7778	0.7078	0.7141	0.6195	0.6856	0.6730	**0.7951**
	NMI	0.2636	0.8000	0.7900	0.7475	0.6655	0.6301	0.6290	0.7399	**0.8001**
	ARI	0.2089	**0.7140**	0.6757	0.5968	0.5876	0.5035	0.5622	0.5757	0.6892
	FMI	0.3903	**0.7618**	0.7298	0.6587	0.6526	0.6343	0.6292	0.6435	0.7487

Table 5. Ablation study on four scRNA-seq datasets

	Pollen				Trachea				Bladder			
Metrics	w/o M	w/o G	w/o C	scDeGNN	w/o M	w/o G	w/o C	scDeGNN	w/o M	w/o G	w/o C	scDeGNN
ACC	0.9043	0.9229	0.9342	**0.9369**	0.9467	0.7310	0.9268	**0.9472**	0.9920	0.7564	0.9980	**0.9992**
NMI	0.9193	0.9410	0.9425	**0.9443**	0.8230	0.4875	0.7899	**0.8247**	0.9630	0.5094	0.9853	**0.9934**
ARI	0.9414	0.9301	0.9374	**0.9456**	0.8783	0.6776	0.8415	**0.8960**	0.9838	0.3231	0.9945	**0.9976**
FMI	0.9281	0.9378	0.9443	**0.9516**	0.9442	0.8268	0.9288	**0.9517**	0.9895	0.6644	0.9970	**0.9987**

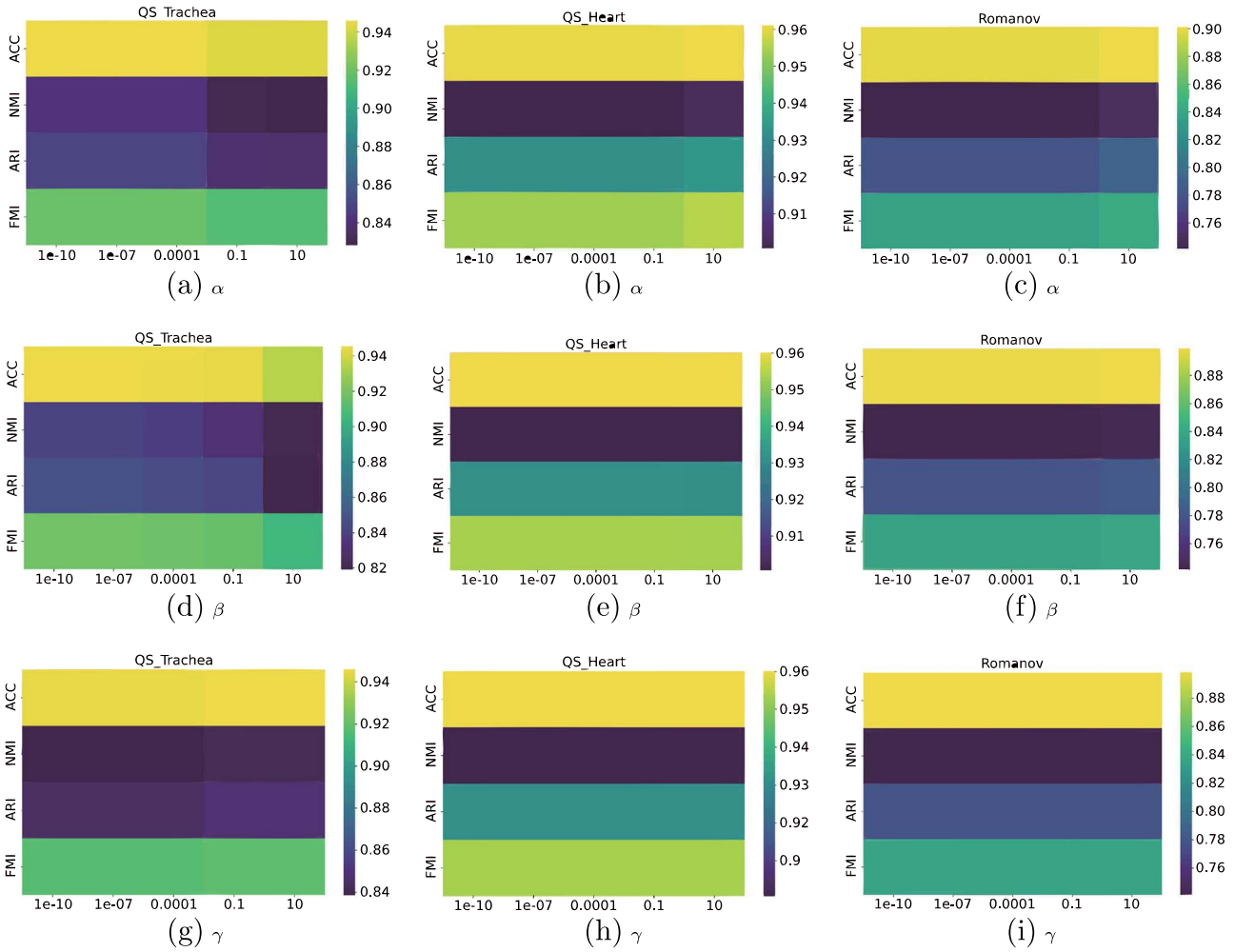

Figure 2. Sensitivity analysis of the parameters α, β, and γ.

on the experimental results shown in Table 5, we can draw the following conclusions:

1. Decoupled GNNs and multi-view learning work synergistically: on the Trachea and Bladder datasets, the removal of either module (w/o M or w/o G) leads to a notable decline in performance. This highlights their complementary contribution. Decoupled GNNs support adaptive feature propagation across views while the diverse biological signals provided by multi-view learning are effectively disentangled by the decoupled architecture.
2. Contrastive learning as a bridge between components: aligning multi-view features through contrastive learning (w/o C) ensures consistency between views, thereby improving the effectiveness of decoupled GNNs. For example, on the Bladder dataset, removing contrastive learning causes the ARI to drop from 99.76% to 99.45%. This indicates that even small misalignments between views can propagate errors through the graph structure, affecting overall performance.

The experimental results reveal that for simpler datasets, such as Pollen, removing certain modules has a minimal effect on performance. However, for more complex hierarchical datasets, such as Trachea and Bladder, removing specific modules results in a significant performance decline (e.g. ARI $\leq$ 67.76%). This indicates that each module in the proposed method contributes to improving clustering performance.

Hyperparameter sensitivity analysis

In the proposed model, there are three hyperparameters: α, β, and γ. Therefore, we conducted experiments to verify the sensitivity of these parameters. Specifically, we set α and β to 0.000001, 0.00001, 0.0001, 0.001, and 0.01, and γ to 0.001, 0.01, 0.1, 1.0, and 10. Figure 2 shows the experimental results of the proposed method on the QS_Trachea, QS_Heart, and Romanov datasets. It can be found that the performance of the proposed model changes when the parameters are varied, but the extent of the change remains within an acceptable range. This indicates that the proposed method is insensitive to these hyperparameters, making it easily applicable to real scRNA-seq data clustering tasks.

It can be seen from Supplementary Figure 2 that the proposed method achieves the best performance on most scRNA-seq datasets when k was set to 20 or 30. With the increase or decrease of the k value, the clustering performance of the proposed method decreases significantly on several datasets. Therefore, we empirically set its value to 20 or 30 in our experiments.

We can see from Supplementary Figure 4 that the proposed model achieved the best performance when the number of layers was set to 2. This indicates that a two-layer GNN effectively balances local and global information extraction while preserving robust representation capability. Additionally, we can see that the proposed method achieves the best results by setting the dimension of the hidden layer to 128 or 256. This suggests that too small a setting may weaken the model's representation capacity,

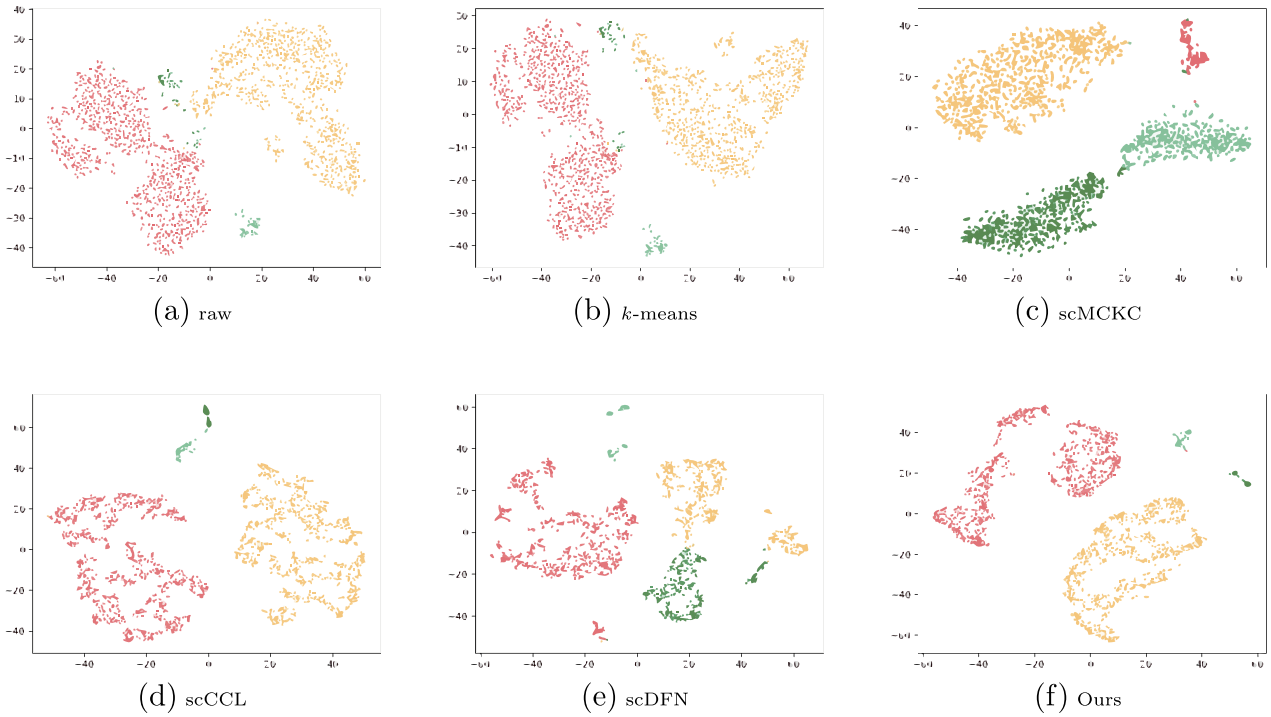

Figure 3. Visualization results of different clustering methods on the Bladder dataset.

Table 6. The computational complexity of various scRNA-seq data clustering methods

Methods	Params(M)	FLOPs(G)
ScCCL	0.444	3.110
scMCKC	6.234	2.091
sc-INDC	2.646	11.436
ADClust	2.395	6.891
DeepScena	6.205	1.647
Ours	**0.033**	**0.004**

whereas an overly large one may introduce redundant information, negatively affecting performance.

Complexity analysis

To assess the complexity of the proposed model, we evaluated it using two metrics: the number of parameters (Params) and floating-point operations (FLOPs). Specifically, Params represents the number of parameters, and FLOPs refers to the number of computations in a model. These two metrics are widely used to measure the complexity of a model. Table 6 shows the experimental results of the proposed method and other competitors. It can be seen that our proposed method requires the fewest parameters and exhibits the lowest computational complexity during training. The efficiency of our approach stems from its streamlined architecture, which incorporates multiple single-layer GNNs and an MLP, resulting in minimal complexity and fewer parameters.

Robustness analysis

In this subsection, we conducted experiments on artificially perturbed datasets to evaluate the robustness of the proposed method. Specifically, we adopted adding and flipping edges to simulate the noise environment on the Bladder datasets. The experimental results of the proposed method on the bladder dataset are shown in Supplementary Tables 1–6. It can be seen that the proposed method can achieve stable clustering performances by adding or flipping 10 to 10 000 edges. Therefore, the proposed method shows strong robustness for adding and flipping operations in scRNA-seq data clustering tasks.

Marker genes analysis

In this subsection, we carried out experiments to assess the biological coherence of the clusters. The experimental results on the Bladder dataset are shown in Supplementary Figure 1. It visualizes gene expression across different cell subtypes, where the x-axis represents gene names and the y-axis represents cell names. We performed clustering analysis on the Bladder dataset using our method to obtain predicted labels, which were then used to identify marker genes and effectively annotate cell types. For instance, we can see from Supplementary Figure 1 that Krt18 is specifically expressed in Cluster 0, which plays a crucial role in urothelial cell (cluster 0) activation and is closely related to bladder cancer progression, metastasis, and therapeutic resistance. Tm4sf1 is uniquely localized to Cluster 3, demonstrating essential functional contributions to endothelial cells.

Visualization

In this experiment, we visualized the scRNA-seq data representations by projecting them into a two-dimensional space. In this study, we performed visualization experiments on the Bladder and QS_Trachea datasets, with the results shown in Figs 3 and 4. It can be seen from Fig. 3 and Fig. 4 that the clustering results of the scCCL and scDFN methods on the Bladder and QS_Trachea datasets reveal the overlap of several cell clusters. In contrast, the proposed scDeGNN method effectively distinguishes each cell cluster in these two datasets, exhibiting superior clustering results.

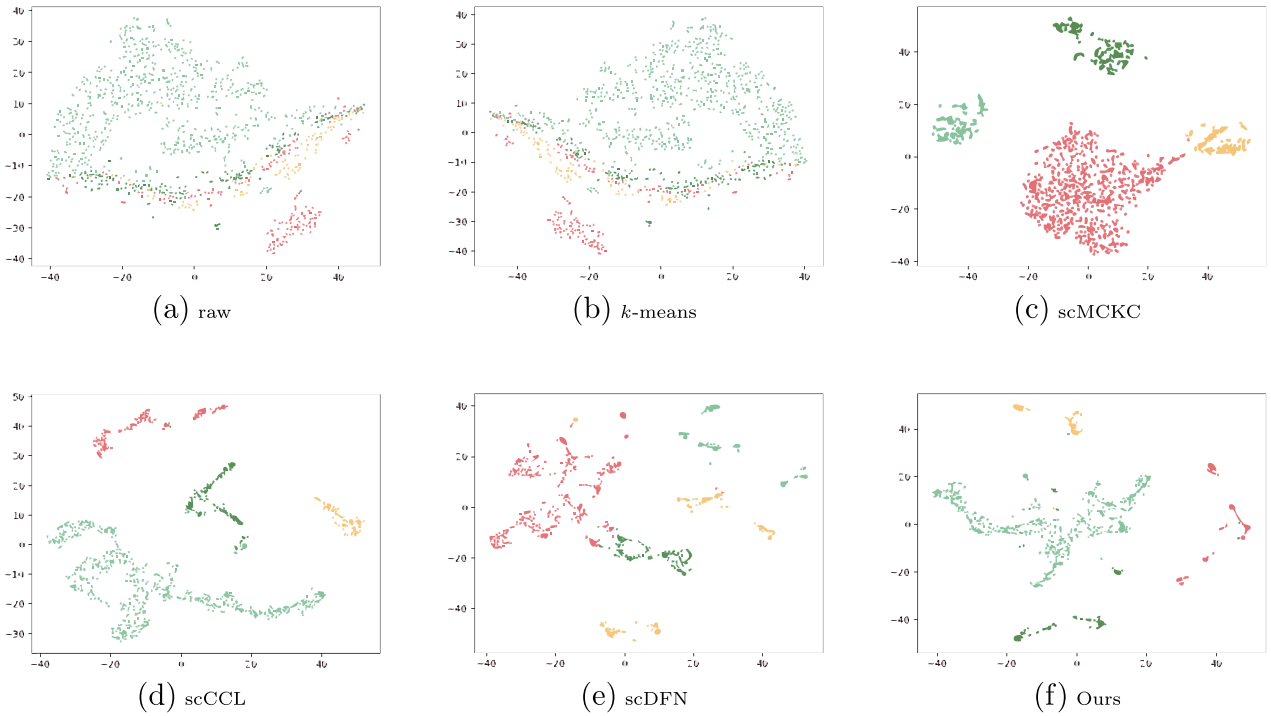

Figure 4. Visualization results on the QS_Trachea dataset.

Discussion and conclusion

In this paper, we propose a multi-view scRNA-seq data clustering method, called decoupled GNNs, based on multi-view contrastive learning (scDeGNN). We construct two distinct adjacency matrices and independently train them using decoupled GNNs, obtaining initial cell representations. They are then passed through an MLP to generate refined cell representations. By merging the representations of multi-view scRNA-seq data, we achieve a more comprehensive feature representation of scRNA-seq data. Finally, we perform the clustering on them and achieve stable clustering results. Extensive experimental results show that the proposed scDeGNN method significantly outperforms other state-of-the-art clustering methods across nine real scRNA-seq data datasets.

Although the proposed method demonstrates promising performance in scRNA-seq data clustering, it has certain limitations requiring further investigation. First, to enhance model robustness and representation quality, we will incorporate adversarial batch alignment methods and feature selection techniques based on Wasserstein stability [55] in our model. Second, we will adapt the proposed method to large-scale scRNA-seq data clustering tasks for high-throughput scenarios. Finally, a multi-omics perspective can be used to construct more biologically meaningful views, enabling a more comprehensive exploration of the multidimensional information of cells.

Key Points

- The proposed multi-view scRNA-seq data learning framework is based on decoupled GNNs, which decompose complex learning tasks into simpler sub-tasks, enabling more efficient network training compared with traditional GCNs.
- The proposed method introduces a multi-view contrastive learning mechanism. On the one hand, it more effectively explores the complementarity of scRNA-seq data in clustering tasks. On the other hand, it enhances the model's robustness and significantly improves its representational capacity by adopting a contrastive learning strategy.
- Experimental results on several benchmark datasets show the effectiveness of the proposed method in scRNA-seq data clustering applications

Supplementary data

Supplementary data is available at *Briefings in Bioinformatics* online.

Conflict of interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Funding

This work was supported by Yunnan Provincial Major Science and Technology Special Plan Projects [Grant No. 202402AD080001].

References

1. Jovic D, Liang X, Zeng H. *et al.* Single-cell RNA sequencing technologies and applications: a brief overview. *Clin Transl Med* 2022;**12**:e694. https://doi.org/10.1002/ctm2.694
2. Ziegenhain C, Vieth B, Parekh S. *et al.* Comparative analysis of single-cell RNA sequencing methods. *Mol Cell* 2017;**65**:631–643.e4. https://doi.org/10.1016/j.molcel.2017.01.023

3. Tiklová K, Björklund ÅK, Lahti L. *et al.* Single-cell RNA sequencing reveals midbrain dopamine neuron diversity emerging during mouse brain development. *Nat Commun* 2019;**10**:581. https://doi.org/10.1038/s41467-019-08453-1
4. Wang Z, Xie X, Liu S. *et al.* scFseCluster: a feature selection-enhanced clustering for single-cell RNA-seq data. *Life Sci Alliance* 2023;**6**:e202302103. https://doi.org/10.26508/lsa.202302103
5. Lee J, Kim S, Hyun D. *et al.* Deep single-cell RNA-seq data clustering with graph prototypical contrastive learning. *Bioinformatics* 2023;**39**:btad342.
6. Von Luxburg U. A tutorial on spectral clustering. *Stat Comput* 2007;**17**:395–416. https://doi.org/10.1007/s11222-007-9033-z
7. Lu C, Yan S, Lin Z. Convex sparse spectral clustering: single-view to multi-view. *IEEE Trans Image Process* 2016;**25**:2833–43. https://doi.org/10.1109/TIP.2016.2553459
8. Xu C, Su Z. Identification of cell types from single-cell transcriptomes using a novel clustering method. *Bioinformatics* 2015;**31**:1974–80. https://doi.org/10.1093/bioinformatics/btv088
9. Van der Maaten L, Hinton G. Visualizing data using t-SNE. *J Mach Learn Res* 2008;**9**:2579–5.
10. Ma S, Dai Y. Principal component analysis based methods in bioinformatics studies. *Brief Bioinform* 2011;**12**:714–22. https://doi.org/10.1093/bib/bbq090
11. Lee DD, Seung HS. Learning the parts of objects by non-negative matrix factorization. *Nature* 1999;**401**:788–91.
12. Wang B, Zhu J, Pierson E. *et al.* Visualization and analysis of single-cell RNA-seq data by kernel-based similarity learning. *Nat Methods* 2017;**14**:414–6. https://doi.org/10.1038/nmeth.4207
13. Wolf FA, Angerer P, Theis FJ. SCANPY: large-scale single-cell gene expression data analysis. *Genome Biol* 2018;**19**:1–5. https://doi.org/10.1186/s13059-017-1382-0
14. Traag VA, Waltman L, Van Eck NJ. From Louvain to Leiden: guaranteeing well-connected communities. *Sci Rep* 2019;**9**:1–12. https://doi.org/10.1038/s41598-019-41695-z
15. Blondel VD, Guillaume JL, Lambiotte R. *et al.* Fast unfolding of communities in large networks. *J Stat Mech: Theory Exp* 2008;**2008**:P10008. https://doi.org/10.1088/1742-5468/2008/10/P10008
16. Shu Z, Long Q, Zhang L. *et al.* Robust graph regularized NMF with dissimilarity and similarity constraints for ScRNA-seq data clustering. *J Chem Inf Model* 2022;**62**:6271–86. https://doi.org/10.1021/acs.jcim.2c01305
17. Eraslan G, Simon LM, Mircea M. *et al.* Single-cell RNA-seq denoising using a deep count autoencoder. *Nat Commun* 2019;**10**:390. https://doi.org/10.1038/s41467-018-07931-2
18. Xie K, Huang Y, Zeng F. *et al.* scAIDE: clustering of large-scale single-cell RNA-seq data reveals putative and rare cell types. *NAR Genomics Bioinf* 2020;**2**:lqaa082.
19. Chen L, Wang W, Zhai Y. *et al.* Single-cell transcriptome data clustering via multinomial modeling and adaptive fuzzy k-means algorithm. *Front Genet* 2020;**11**:295. https://doi.org/10.3389/fgene.2020.00295
20. Geddes TA, Kim T, Nan L. *et al.* Autoencoder-based cluster ensembles for single-cell RNA-seq data analysis. *BMC Bioinformatics* 2019;**20**:1–11. https://doi.org/10.1186/s12859-019-3179-5
21. Wang J, Xia J, Wang H. *et al.* scDCCA: deep contrastive clustering for single-cell RNA-seq data based on auto-encoder network. *Brief Bioinform* 2023;**24**:bbac625.
22. Li X, Wang K, Lyu Y. *et al.* Deep learning enables accurate clustering with batch effect removal in single-cell RNA-seq analysis. *Nat Commun* 2020;**11**:2338. https://doi.org/10.1038/s41467-020-15851-3
23. Meng R, Yin S, Sun J. *et al.* scAAGA: single cell data analysis framework using asymmetric autoencoder with gene attention. *Comput Biol Med* 2023;**165**:107414. https://doi.org/10.1016/j.compbiomed.2023.107414
24. Zeng Y, Wei Z, Zhong F. *et al.* A parameter-free deep embedded clustering method for single-cell RNA-seq data. *Brief Bioinform* 2022;**23**:bbac172.
25. Hartigan JA, Hartigan PM. The dip test of unimodality. *Ann Stat* 1985;70–84.
26. He Y, Chen X, Tu NH. *et al.* Deep multi-constraint soft clustering analysis for single-cell RNA-seq data via zero-inflated autoencoder embedding. *IEEE/ACM Trans Comput Biol Bioinform* 2023;**20**:2254–65. https://doi.org/10.1109/TCBB.2023.3240253
27. Lei T, Chen R, Zhang S. *et al.* Self-supervised deep clustering of single-cell RNA-seq data to hierarchically detect rare cell populations. *Brief Bioinform* 2023;**24**:bbad335.
28. Mondal AK, Joshi I, Singh P. *et al.* Clustering single-cell RNA sequence data using information maximized and noise-invariant representations. *IEEE/ACM Trans Comput Biol Bioinform* 2022;**20**:1983–94. https://doi.org/10.1109/TCBB.2022.3227202
29. Wang T, Li B, Nabavi S. Single-cell RNA sequencing data clustering using graph convolutional networks. In: *2021 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pp. 2163–70, Houston, TX, USA: IEEE, 2021.
30. Li J, Jiang W, Han H. *et al.* ScGSLC: an unsupervised graph similarity learning framework for single-cell RNA-seq data clustering. *Comput Biol Chem* 2021;**90**:107415. https://doi.org/10.1016/j.compbiolchem.2020.107415
31. Wang J, Ma A, Chang Y. *et al.* scGNN is a novel graph neural network framework for single-cell RNA-seq analyses. *Nat Commun* 2021;**12**:1882. https://doi.org/10.1038/s41467-021-22197-x
32. Peng H, Fan W, Fang C. *et al.* SCMAG: a semisupervised single-cell clustering method based on matrix aggregation graph convolutional neural network. *Comput Math Methods Med* 2021;**2021**:1–6. https://doi.org/10.1155/2021/6842752
33. Wang HY, Zhao JP, Su YS. *et al.* scCDG: a method based on DAE and GCN for scRNA-seq data analysis. *IEEE/ACM Trans Comput Biol Bioinform* 2021;**19**:3685–94. https://doi.org/10.1109/TCBB.2021.3126641
34. Risso D, Perraudeau F, Gribkova S. *et al.* ZINB-WaVE: a general and flexible method for signal extraction from single-cell RNA-seq data. *BioRxiv* 2017;125112.
35. Liu T, Jia C, Bi Y. *et al.* scDFN: enhancing single-cell RNA-seq clustering with deep fusion networks. *Brief Bioinform* 2024;**25**:bbae486.
36. Hu D, Dong Z, Liang K. *et al.* Single-cell multi-view clustering via community detection with unknown number of clusters arXiv preprint arXiv:231117103. 2023.
37. Du ZH, Hu WL, Li JQ. *et al.* scPML: pathway-based multi-view learning for cell type annotation from single-cell RNA-seq data. *Commun Biol* 2023;**6**. https://doi.org/10.1038/s42003-023-05634-z
38. Sun H, Qu H, Duan K. *et al.* scMGCN: a multi-view graph convolutional network for cell type identification in scRNA-seq data. *Int J Mol Sci* 2024;**25**:2234. https://doi.org/10.3390/ijms25042234
39. Du L, Han R, Liu B. *et al.* ScCCL: single-cell data clustering based on self-supervised contrastive learning. *IEEE/ACM Trans Comput Biol Bioinform* 2023;**20**:2233–41. https://doi.org/10.1109/TCBB.2023.3241129
40. Shu Z, Xia M, Tan K. *et al.* Multi-level multi-view network based on structural contrastive learning for scRNA-seq data clustering. *Brief Bioinform* 2024;**25**:bbae562.
41. Peterson LE. K-nearest neighbor. *Scholarpedia* 2009;**4**:1883. https://doi.org/10.4249/scholarpedia.1883

42. Coifman RR, Lafon S. Diffusion maps. *Appl Comput Harmon Anal* 2006;**21**:5–30. https://doi.org/10.1016/j.acha.2006.04.006
43. Wu F, Souza A, Zhang T. *et al.* Simplifying graph convolutional networks. In: *International conference on machine learning*, pp. 6861–71. Long Beach, CA, USA: PMLR, 2019.
44. Zhang H, Zhu Y, Li X. Decouple graph neural networks: train multiple simple GNNs simultaneously instead of one. *IEEE Trans Pattern Anal Mach Intell* 2024.
45. Cai D, He X, Han J. Document clustering using locality preserving indexing. *IEEE Trans Knowl Data Eng* 2005;**17**:1624–37. https://doi.org/10.1109/TKDE.2005.198
46. Santos JM, Embrechts M. On the use of the adjusted rand index as a metric for evaluating supervised classification. In: *International conference on artificial neural networks*, pp. 175–84. Limassol, Cyprus: Springer, 2009.
47. Strehl A, Ghosh J. Cluster ensembles-a knowledge reuse framework for combining multiple partitions. *J Mach Learn Res* 2002;**3**:583–617.
48. Fowlkes EB, Mallows CL. A method for comparing two hierarchical clusterings. *J Am Stat Assoc* 1983;**78**:553–69. https://doi.org/10.1080/01621459.1983.10478008
49. Schaum N, Karkanias J, Neff NF. *et al.* Single-cell transcriptomics of 20 mouse organs creates a tabula Muris: the tabula Muris consortium. *Nature* 2018;**562**:367–72. https://doi.org/10.1038/s41586-018-0590-4
50. Han Y. *Pollen Dataset in Semiarid Region of Northern China*. Peking University Open Research Data Platform; 2020. Available from: https://doi.org/10.18170/DVN/V9JN52
51. Chen R, Wu X, Jiang L. *et al.* Single-cell RNA-seq reveals hypothalamic cell diversity. *Cell Rep* 2017;**18**:3227–41.
52. Romanov RA, Zeisel A, Bakker J. *et al.* Molecular interrogation of hypothalamic organization reveals distinct dopamine neuronal subtypes. *Nat Neurosci* 2017;**20**:176–88. https://doi.org/10.1038/nn.4462
53. Young MD, Mitchell TJ, Vieira Braga FA. *et al.* Single-cell transcriptomes from human kidneys reveal the cellular identity of renal tumors. *Science* 2018;**361**:594–9. https://doi.org/10.1126/science.aat1699
54. Hartigan JA, Wong MA. Algorithm AS 136: a k-means clustering algorithm. *J R Stat Soc Ser C Appl Stat* 1979;**28**:100–8.
55. Heumos L, Schaar AC, Lance C. *et al.* Best practices for single-cell analysis across modalities. *Nat Rev Genet* 2023;**24**:550–72. https://doi.org/10.1038/s41576-023-00586-w

Briefings in Bioinformatics, 2025, **26(3)**, bbaf198
https://doi.org/10.1093/bib/bbaf198
Problem Solving Protocol