scientific reports

OPEN

A robust multi-scale clustering framework for single-cell RNA-seq data analysis

Songrun Jiang, Chunyan Wang✉, Qiucheng Sun✉ & Zhi Zhang

Recent advancements in single-cell RNA sequencing (scRNA-seq) technology have unlocked novel opportunities for deep exploration of gene expression patterns. However, the inherent high dimensionality, sparsity, and noise in scRNA-seq data pose significant challenges for existing clustering methods, especially in accurately identifying and classifying diverse cell types. To address these challenges, we introduce a new method, single-cell Multi-Scale Clustering Framework (scMSCF), which combines multi-dimensional PCA for dimensionality reduction, K-means clustering, and a weighted ensemble meta-clustering approach, enhanced by a self-attention-driven Transformer model to optimize clustering performance. scMSCF constructs an initial clustering framework using a multi-layer dimensionality reduction strategy to establish a robust consensus on clustering structure. A voting mechanism within the meta-clustering process selects high-confidence cells from the initial clustering results to provide precise training labels for the Transformer model. This approach enables the model to capture complex dependencies in gene expression data, thereby enhancing clustering accuracy. Comprehensive testing across eight single-cell RNA sequencing datasets demonstrates that scMSCF surpasses existing methods, achieving on average 10-15% higher ARI, NMI, and ACC scores. For example, on the PBMC5k dataset, scMSCF improves ARI from 0.72 to 0.86, demonstrating its ability to accurately identify diverse cell populations. The source code for our algorithm is publicly available at https://github.com/DEREKJ24/scMSCF.

Keywords scRNA-seq, Multi-dimensional PCA, Weighted ensemble meta-clustering, Transformer model, High-confidence cells

Single-cell RNA sequencing (scRNA-seq) technology offers unprecedented opportunities for modern biological research by capturing subtle variations in gene expression at the single-cell level[1]. This technology has played a crucial role in uncovering cellular heterogeneity, studying disease mechanisms, and exploring cell development processes[2,3]. However, the high sparsity, noise, and dimensionality of scRNA-seq data pose significant challenges for data processing and analysis. Clustering analysis has become an essential tool for interpreting and revealing cell types and functional states within these complex datasets. Through clustering analysis, scientists can identify groups of cells with similar expression patterns from gene expression data, thus gaining deeper insights into the cellular composition of tissues and their dynamic changes in biological processes and disease states[4].

In the analysis of scRNA-seq data, early clustering methods, such as K-means and hierarchical clustering, performed well on low-dimensional datasets but often struggled with high-dimensional, sparse, and noisy data[5]. K-means relies on Euclidean distances, making it sensitive to noise and uneven data distributions, while hierarchical clustering, although capable of revealing data structure, tends to be unstable in high-dimensional spaces[6]. Model-based clustering methods, such as the Gaussian Mixture Model (GMM), attempt to handle complex clustering tasks by assuming data distributions, yet these methods are sensitive to parameter choices and computationally expensive[7]. Spectral clustering[8] improves clustering performance by leveraging the spectral properties of data but lacks efficiency in handling large-scale datasets. Common issues with these methods include assumptions of uniform low-dimensional data distribution, neglect of data sparsity and noise, and limitations in high-dimensional space, which restrict their utility in scRNA-seq data analysis, especially in distinguishing biologically distinct yet similar cell groups[9].

To more effectively address these challenges, some emerging methods have become popular. Seurat[10] integrates normalization, scaling, and PCA to preprocess data, constructs a cell similarity graph via k-nearest neighbors, and applies Louvain or Leiden algorithms for community detection. Phenograph[11] constructs cell

College of Computer Science and Technology, Changchun Normal University, Changchun 130000, China. ✉email: wangchunyan@ccsfu.edu.cn; sunqiucheng@ccsfu.edu.cn

relationships using k-nearest neighbors and applies community detection algorithms to discover cell groups, making it particularly suitable for biologically related cell types with subtle expression pattern differences. Additionally, the combined use of classical algorithms has become a popular approach for effective clustering, such as combining T-SNE[12] and DBSCAN[13], leveraging their strengths to improve clustering accuracy. T-SNE maps high-dimensional data to lower-dimensional space while preserving point similarities, enabling DBSCAN to detect clusters based on connectivity of core and boundary points, which is effective for datasets with well-defined cluster boundaries.

Despite the effectiveness of these methods in improving clustering performance, a single clustering scheme often faces limitations due to algorithmic assumptions, parameter choices, and sensitivity to data characteristics. Since individual algorithms struggle to fully capture data diversity and complexity, relying solely on one approach may overlook crucial biological features. To address this issue, methods such as SIMLR[14] create data similarity metrics using multi-kernel learning, SC3[15] utilizes multiple clustering results to enhance clustering decisions, and SHARP[16] uses ensemble strategies based on random projections to improve clustering speed and accuracy, especially for large and noisy datasets. These approaches aim to leverage the advantages of various clustering results to reduce biases and errors from individual methods. However, integrating clustering results frequently requires running multiple algorithms or configurations, increasing computational costs, particularly with large datasets. Additionally, choosing and weighting clustering results are critical; improper handling may degrade clustering quality, suggesting that while integration strategies can enhance stability and robustness, careful application is needed to ensure clustering accuracy and practicality.

In recent years, deep learning[17] has been widely applied to scRNA-seq clustering due to its exceptional ability to handle nonlinear and high-dimensional data[18]. Unsupervised learning[19] excels in discovering cellular structures and classifications from unlabeled data, making it indispensable for scRNA-seq clustering. For example, scDSC[20] employs an autoencoder architecture to extract high-dimensional features from gene expression data and combines spectral clustering to enhance the identification of cell groups with diverse expression patterns, enabling precise separation and classification of cell subpopulations. scMAE[21] uses a masking strategy[22] for feature learning, successfully capturing complex features in cell expression to improve clustering performance. CellVGAE[23] combines graph neural networks[24] and variational autoencoders[25] to model cell relationships, providing more accurate clustering for single-cell data. DESC[26] optimizes clustering objectives iteratively, gradually removing batch effects in data with minimal technical differences. Semi-supervised[27] learning leverages a small amount of labeled data to guide model training, effectively supporting cell type identification and classification without relying solely on labeled data. This approach has gained significant attention from researchers, as it introduces limited label information to enhance model performance and address challenges in scenarios with insufficient annotations. scSemiAAE[28] integrates an autoencoder with ZINB[29] loss, adversarial training[30], and a semi-supervised module to achieve better latent representations, enhancing clustering accuracy and scalability. scTPC[31] combines a ZINB-based denoising autoencoder with label information, using deep clustering and weighted cross-entropy[32] to improve clustering accuracy for scRNA-seq data. scSSA[33] integrates semi-supervised autoencoders with FastICA[34] for dimensionality reduction, employing Gaussian mixture models[35] with the BIC index[36] to cluster rare cell subtypes and similar cell types. LFSC[37] combines reference samples to generate a dictionary matrix[38] and anchor graphs[39] to maintain clustering structures, significantly improving clustering efficiency and effectiveness. Although deep learning models perform well in scRNA-seq clustering, the proper division of training and validation sets is critical for model generalization and clustering precision. Deep models rely heavily on high-quality training data to distinguish subtle differences in complex biological data. Improper division can lead to model overfitting, reducing adaptability to new data and hindering the capture of true biological features. Thus, a well-structured and information-rich training and validation set not only ensures model stability and accuracy but also serves as the foundation for generalization, enabling efficient and precise clustering across diverse scRNA-seq datasets. As summarized in Table 1, existing methods for scRNA-seq clustering exhibit methodological diversity but share limitations such as dependency on preset parameters, computational inefficiency, or limited generalizability. Extended feature and applicability contrasts of other approaches are provided in Supplementary Document ST1.

Therefore, we present scMSCF, a novel algorithm based on three core steps to enhance clustering accuracy and stability in single-cell RNA sequencing data. First, scMSCF applies a multi-dimensional PCA strategy for dimensionality reduction, performing K-means clustering across each dimension, and integrates these results through a weighted meta-clustering method to effectively manage data sparsity and noise. Next, a voting mechanism selects high-confidence cells with consistent clustering results to form a stable and reliable training set, providing precise labels for deep learning. Finally, scMSCF incorporates a Transformer model[40] with self-attention to capture complex dependencies in gene expression data, thereby achieving more nuanced cell type

Method	Technical strategy	Advantages	Use cases
scDSC	Autoencoder + Spectral Clustering	Robust feature extraction; handles non-linear structures	Clear-cluster-boundary data
scMAE	Masked Autoencoder	Robust to dropout noise via random masking	Sparse large-scale data
CellVGAE	GNN + Variational Autoencoder	Captures local cell relationships; good for microenvironments	Small-scale fine-grained data
scSemiAAE	Semi-supervised AE + Adversarial Training	Leverages limited labeled data; stable training	Partially labeled data
LFSC	Anchor Graph + Dictionary Learning	Efficient integration across batches or species	Cross-dataset analysis with references
scMSCF	**Multi-PCA + Transformer**	**Confidence-based training; High clustering accuracy**	**High-noise, heterogeneous data**

Table 1. Comparative analysis of technical strategies, advantages, and use cases in scRNA-seq clustering.

classification and relationship modeling. Overall, by integrating multi-dimensional reduction, weighted meta-clustering, and the Transformer model, scMSCF overcomes challenges in high dimensionality, heterogeneity, and sparsity, providing a robust framework for scRNA-seq analysis. Subsequent sections will detail scMSCF's implementation, key steps, and performance across various datasets, showcasing its potential in single-cell RNA sequencing data analysis.

Materials and methods

Overview of scMSCF

The scMSCF framework is a multi-module clustering algorithm specifically designed for scRNA-seq data to enhance clustering accuracy and stability. The workflow of scMSCF incorporates critical modules, such as data preprocessing, dimensionality reduction, clustering, meta-clustering integration, and deep learning optimization, as depicted in Fig. 1. The algorithm takes a single-cell RNA count matrix as input, where each row represents a unique cell and each column represents a gene. The preprocessing module includes quality control, normalization, and batch effect correction to ensure data consistency.

Data preprocessing (Moudle 1)

To enhance dimensionality reduction and clustering efficiency, we applied SCTransform normalization in Seurat v4.3.0, followed by the selection of the top 2000 highly variable genes (HVGs). SCTransform utilizes regularized negative binomial regression to normalize the count data while mitigating sequencing depth and technical noise. This method provides improved variance stabilization compared to traditional log-normalization. The selected HVGs were then used for principal component analysis (PCA), forming the basis for downstream clustering analysis.

In addition to this feature selection-based approach, we also applied SCTransform normalization to the entire gene expression dataset without HVG selection. This ensures that all biological signals are retained, allowing the Transformer model to effectively capture intricate gene-gene relationships and uncover complex transcriptional patterns. Further details on this preprocessing strategy, including parameter settings and validation experiments, are provided in Supplementary Document SN1.

Initial consensus clustering framework (Moudle 2)

Step 1: Multi-dimensional PCA dimensionality reduction and K-means clustering After data preprocessing, although technical noise and batch effects have been effectively mitigated, the high dimensionality of gene expression data can still lead to issues such as the "curse of dimensionality," data sparsity, and increased computational complexity. To address these challenges, we used principal component analysis (PCA) to reduce the dimensionality of the gene expression matrix. PCA is a commonly used linear dimensionality reduction technique that projects high-dimensional data into a lower-dimensional space while preserving as much variance as possible. This simplifies the data structure and reduces redundancy. Let X be the preprocessed gene expression matrix with selected highly variable genes, where n is the number of cells and p is the number of genes. First, we construct the covariance matrix of the gene expression data:

$$\sum = \frac{1}{\mathrm{n}-1}(\mathrm{X}-\bar{\mathrm{X}})^{\mathrm{T}}(\mathrm{X}-\bar{\mathrm{X}}) \tag{1}$$

where $\mathrm{X}-\bar{\mathrm{X}}$ is the gene expression matrix after mean centering. Subsequently, the covariance matrix is eigen-decomposed to extract the eigenvectors corresponding to the top k largest eigenvalues, and these are assembled into the matrix $\mathrm{V_k}$. Ultimately, the gene expression matrix is mapped into a reduced k-dimensional space:

$$\mathrm{X_{reduced}} = \mathrm{XV_k} \tag{2}$$

In this study, we adopt a multi-resolution dimensionality reduction strategy, referred to as multi-dimensional PCA. Specifically, this term denotes the application of PCA multiple times using different numbers of retained principal components: 30, 35, 40, 45, and 50. Unlike traditional approaches that perform clustering on a fixed-dimensional PCA projection (e.g., 50 components), our approach explores the data structure from multiple abstraction levels. This strategy allows us to capture distinct aspects of the underlying cellular heterogeneity that may be emphasized at different projection depths.

After dimensionality reduction under each selected dimension, we applied the K-means algorithm to perform clustering. Given a predefined number of clusters K, the algorithm iteratively updates the cluster centers μ and assigns data points x to their corresponding clusters C to minimize the following objective function:

$$\min_{\mathrm{C_1,\ldots,C_k}} \sum_{\mathrm{i=1}}^{\mathrm{k}} \sum_{\mathrm{x}\in \mathrm{C_i}} \|\mathrm{x}-\mu_{\mathrm{i}}\|^2 \tag{3}$$

where $\mathrm{C_i}$ represents the i-th cluster, and μ_{i} is the mean of all points x in $\mathrm{C_i}$. We applied the elbow method to determine the optimal number of clusters, selecting two adjacent K values at each dimension for clustering to ensure robustness. For details on the application of the elbow method and multi-dimensional PCA dimensionality reduction, please refer to Supplementary Document SN2.

Step 2: Weighted integration of multiple clustering results To boost both the accuracy and stability of the clustering outcomes, our study employed a weighted meta-clustering approach that integrates multiple

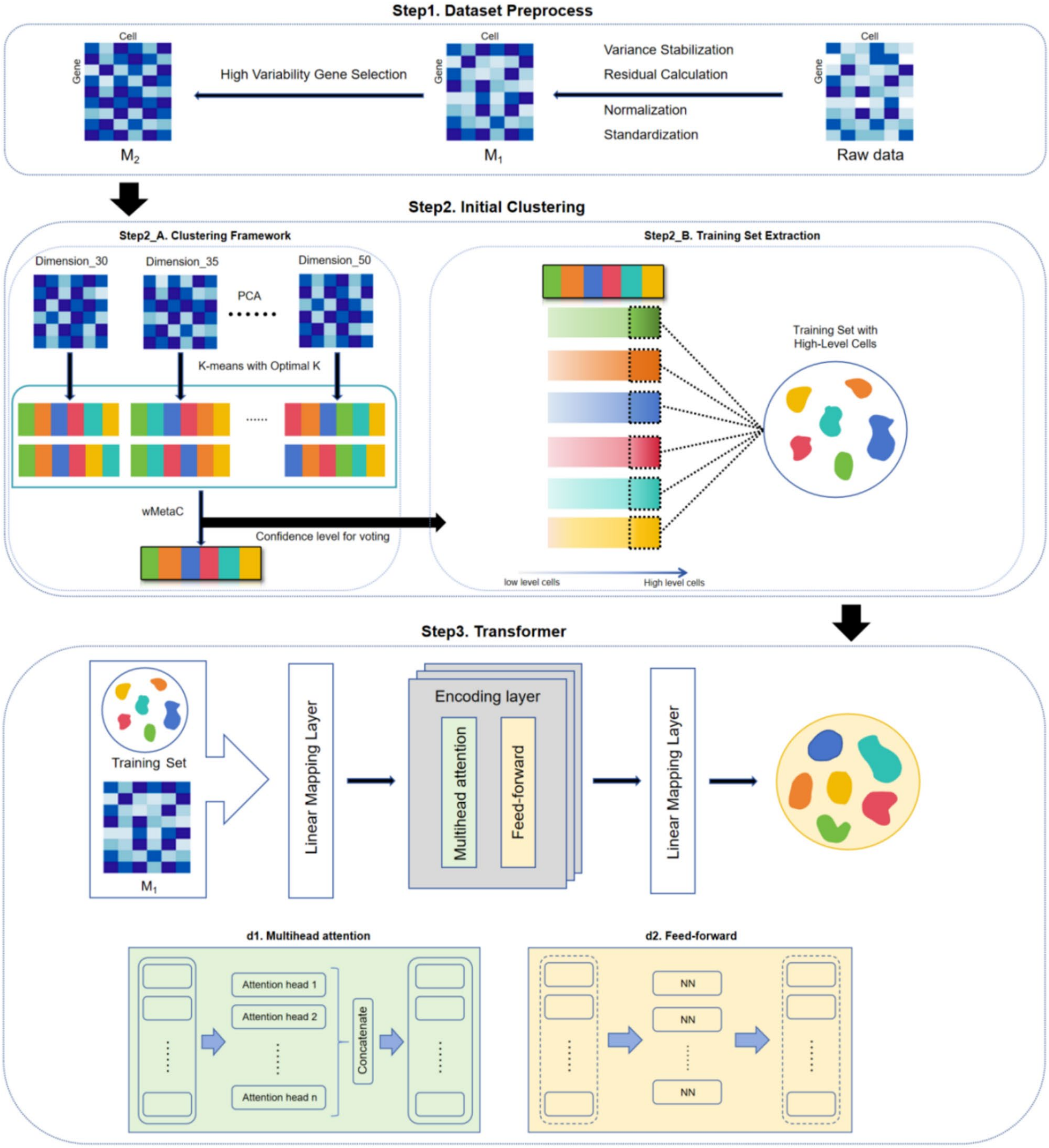


Fig. 1. The overall scMSCF workflow. scMSCF starts with preprocessing single-cell RNA sequencing data, where SCTransform normalizes the data, followed by the selection of highly variable genes. PCA is then applied for dimensionality reduction to target dimensions of 30, 35, 40, 45, and 50. At each dimension, K-means clustering is conducted by clustering two adjacent points (representing two k-values) on the elbow curve, generating multiple clustering results. These results are then integrated into an initial clustering framework using a weighted ensemble meta-clustering (wMetaC) approach. Additionally, high-confidence cells are selected as the training set through a voting mechanism within wMetaC based on confidence calculations, and a Transformer model is introduced for deep learning. The self-attention mechanism of the Transformer learns complex patterns in the data, enhancing the accuracy and stability of cell clustering.

clustering results[16]. This method strengthens the final clustering performance by assigning weights based on the consistency of cell groupings across different clustering results. Specifically, we calculated a weight for each cell that reflects how consistently it appears in the same cluster across multiple results. Cells that frequently appear together in the same cluster receive a higher weight, whereas cells that are inconsistently grouped across clusters are assigned lower weights. The weight calculation is given by:

$$\mathrm{w_{ij}} = \mathrm{s_{ij}} \times (1 - \mathrm{s_{ij}}) \tag{4}$$

where $\mathrm{s_{ij}}$ denotes the similarity between cells i and j across different clustering results. To better illustrate this process, consider a simple example: suppose we have three clustering results for five cells (C1 to C5). If cells C1 and C2 appear together in the same cluster in two out of three clustering results, they are considered highly similar, resulting in a higher similarity score and thus a lower weight. Conversely, if cells C1 and C5 never appear in the same cluster, their similarity score is lower, resulting in a higher weight. These similarity-derived weights help emphasize inconsistently clustered cells and strengthen the consensus of stable cell relationships across multiple runs. Utilizing these weights, we computed a weighted pairwise similarity matrix, which provides a more accurate measure of relationships between clusters by incorporating cell-specific weights. The weighted similarity calculation is as follows:

$$\mathrm{S_{wMetaC}} = \frac{\sum_{\mathrm{t} \in \mathrm{C_u} \cap \mathrm{C_v}} \sum_{\mathrm{j}=1}^{\mathrm{N}} \mathrm{W_{t,j}} + \delta}{\sum_{\mathrm{t} \in \mathrm{C_u} \cup \mathrm{C_v}} \sum_{\mathrm{j}=1}^{\mathrm{N}} \mathrm{W_{t,j}} + \delta} \tag{5}$$

where $\mathrm{C_u}$ and $\mathrm{C_v}$ denote two distinct clusters, each representing a separate group of single cells identified in the clustering results. The variable t indexes each individual cell within cluster C, iterating over all such cells; $\mathrm{W_{t,j}}$ represents the pairwise weight between cell t and cell j, reflecting their similarity. N is the total number of cells, used to iterate over all cells for summing pairwise weights, and δ is a small positive constant added to prevent a zero denominator, ensuring formula stability. After constructing the weighted similarity matrix for the clusters, hierarchical clustering is applied to further consolidate the clusters. This involves calculating the distance between all clusters and merging the most similar ones to yield a more robust clustering structure. Finally, a voting mechanism determines each cell's final cluster assignment: each cell 'votes' based on its cluster assignments across multiple clustering results, and the cluster with the highest votes is assigned as its final cluster.

Through this weighted meta-clustering strategy, we successfully integrate multiple clustering results. A detailed illustrative diagram of this process is provided in Supplementary Document SN3.

Training and validation set division based on voting confidence (Moudle 3)

To further enhance clustering accuracy, we implemented a deep learning model designed to capture complex data dependencies. Before model implementation, it was crucial to strategically divide the training and validation sets to enable key feature extraction from high-quality data. scMSCF leverages a voting mechanism from weighted meta-clustering to identify high-confidence cells, ensuring the training data is accurately labeled. Specifically, scMSCF calculates each cell's confidence level in each cluster based on multiple clustering results. Each cell's cluster assignments across ten different rounds of dimensionality reduction and K-means clustering are recorded, and the number of votes it receives in each cluster is counted. A higher vote count in a specific cluster reflects a higher confidence level, indicating that the cell is more likely to belong to that cluster. The formula for calculating confidence in a cell's assigned cluster is as follows:

$$\mathrm{Confidence_i} = \frac{\mathrm{Votes_i}}{\mathrm{Total\ Votes}} \tag{6}$$

where $\mathrm{Votes_i}$ denotes the number of times i cell was assigned to its final cluster, and $\mathrm{Total\ Votes}$ refers to the total number of clustering rounds (10 in this study). This voting-based confidence calculation ensures that each cell's final cluster assignment is the one with the highest consistency across multiple clustering results, offering enhanced stability and reliability. Based on these confidence scores, cells with high confidence are allocated in the training set, and those with lower confidence to the validation set. High-confidence cells are selected for training because they show consistent clustering across multiple results, providing stable and reliable labels that enable the model to learn features more effectively.

Deep learning enhancement: incorporating the transformer model (Moudle 4)

Following the division of the training and validation sets, we constructed a Transformer model using the PyTorch framework and proceeded with in-depth training and prediction. Equipped with a self-attention mechanism, the Transformer model excels at capturing long-range dependencies and intricate patterns within gene expression data. In recent years, it has achieved remarkable success in natural language processing and is increasingly being applied in bioinformatics. Initially, the input data is projected into a specified dimension via a linear layer, which is immediately followed by layer normalization:

$$\mathrm{h_0} = \mathrm{LayerNorm}\left(\mathrm{W_{input}} \mathrm{x} + \mathrm{b_{input}}\right) \tag{7}$$

where $\mathrm{W_{input}}$ and $\mathrm{b_{input}}$ denote the weight and bias parameters of the input layer, respectively, with x representing the input feature vector and $\mathrm{h_0}$ as the initial hidden state. The input data then passes through multiple layers of the Transformer encoder, each comprising a multi-head self-attention mechanism and a feedforward neural network. The self-attention mechanism is computed as follows:

$$\mathrm{Atention}\left(\mathrm{Q},\mathrm{K},\mathrm{V}\right)=\mathrm{softmax}\left(\frac{\mathrm{QK}^{\mathrm{T}}}{\sqrt{\mathrm{d}_{\mathrm{k}}}}\right)\mathrm{V} \tag{8}$$

where Q, K, V represent the input matrices derived from linear transformations, and d_{k} denotes the dimensionality of the key vectors. The multi-head attention mechanism concatenates the outputs from multiple self-attention heads as follows:

$$\mathrm{MultiHead}\left(\mathrm{Q}\ ,\ \mathrm{K},\mathrm{V}\right)=\mathrm{Contact}\left(\mathrm{head}_{1},\ \mathrm{head}_{2},\ldots,\ \mathrm{head}_{\mathrm{h}}\right)\mathrm{W}_{\mathrm{O}} \tag{9}$$

where $\mathrm{head}_{\mathrm{h}}$ denotes the output of the h-th self-attention head, and W_{O} is the weight matrix of the output layer. The concatenated output is then further processed by a feedforward neural network:

$$\mathrm{FFN}\left(\mathrm{h}\right)=\max\left(0,\mathrm{W}_{1}\mathrm{h}+\mathrm{b}_{1}\right)\mathrm{W}_{2}+\mathrm{b}_{2} \tag{10}$$

Where W_{1} and W_{2} denote the weight matrices, and b_{1} and b_{2} represent the bias vectors of the feedforward network, with h_{L} as the hidden state. Finally, a linear classifier generates the clustering label for each cell:

$$\widehat{\mathrm{y}}=\mathrm{softmax}\left(\mathrm{W}_{\mathrm{out}}\mathrm{h}_{\mathrm{L}}+\mathrm{b}_{\mathrm{out}}\right) \tag{11}$$

Where $\mathrm{W}_{\mathrm{out}}$ and $\mathrm{b}_{\mathrm{out}}$ represent the weight matrix and bias vector of the output layer, respectively, h_{L} denotes the output from the final encoder layer, and $\widehat{\mathrm{y}}$ corresponds to the model's predicted outputs. During training, the model parameters are optimized using the cross-entropy loss function and the Adam optimizer. The cross-entropy loss function is defined as follows:

$$\mathrm{L}=-\sum_{\mathrm{i}=1}^{\mathrm{N}}\mathrm{y}_{\mathrm{i}}\log\left(\mathrm{p}_{\mathrm{i}}\right) \tag{12}$$

Where N denotes the sample size, y_{i} is the true label, and p_{i} represents the model-predicted probability. Throughout the training phase, optimization of model parameters is performed using the cross-entropy loss function and the Adam optimizer.

To prevent overfitting and ensure robust performance on unseen data, we evaluated model performance on the validation set and employed an early stopping strategy. If validation loss fails to decrease over consecutive training epochs, training halts, and the best model to that point is preserved. This early stopping strategy further enhances the model's generalization capability. By integrating accuracy metrics with early stopping, we achieve a more comprehensive evaluation of the model's performance on new data. Once training is complete, the trained Transformer model is used to predict clustering labels for unlabeled gene expression data, which supports further biological analysis and applications. The detailed parameter settings and tuning specifications for the Transformer model are provided in supplementary document SN4.

Datasets

To assess the performance of the scMSCF method, we conducted comparative experiments using eight publicly available single-cell RNA sequencing datasets. Table 2 presents detailed information on these datasets, including the number of cells, the number of genes, the number of cell subtypes, and the source organ/tissue type of each dataset. To ensure the broad applicability of the proposed method, we selected multiple datasets of varying scales and those with diverse dimensional characteristics of gene expression, with the aim of evaluating the effectiveness of scMSCF across different types of datasets.

Dataset	Organ/Tissue	Cells	Genes	Type	Repositories	Accession numbers
Mouse Kidney Nuclei	Mouse kidney	1385	32,285	9	10X Genomics	
PBMCs - v3	Peripheral blood	4352	33,694	9	10X Genomics	
Hodgkin's Lymphoma Targeted-Compare	Lymph node tumor	2913	1056	8	10X Genomics	
5k Healthy Donor PBMCs	Peripheral blood	5025	33,538	8	10X Genomics	
PBMCs - Chromium Connect	Peripheral blood	3363	33,538	6	10X Genomics	
BLA Fear Memory	Basolateral amygdala	6361	44,272	7	Gene expression omnibus	GSE246147
Lymphoma-TIMs	Tumor-Infiltrating myeloid cells	615	11,283	7	Gene expression omnibus	GSE154763
iPSC Midbrain Neuron PD	Midbrain DA Neurons	767	13,402	7	Gene expression omnibus	GSE247600

Table 2. Summary of the real single-cell RNA-seq datasets.

Results

Clustering performance analysis and comparison with other clustering methods

To comprehensively evaluate the clustering performance of scMSCF in identifying cell types, we compared it against nine widely recognized clustering methods for single-cell RNA sequencing (scRNA-seq) data. These methods represent a diverse range of strategies, including consensus clustering, graph-based techniques, dimensionality reduction approaches, and deep learning models, each selected for its established significance and unique advantages in scRNA-seq analysis. The performance of each algorithm was assessed using five standard evaluation metrics: Adjusted Rand Index (ARI)[41], Normalized Mutual Information (NMI)[42], Clustering Accuracy (ACC)[43], Variation of Information (VI)[44], and Davies-Bouldin Index (DBI)[45]. Together, these metrics provide a robust framework for an objective and multidimensional analysis of clustering performance, capturing key aspects such as consistency, information retention, and structural integrity. This comprehensive comparison underscores scMSCF's robustness, versatility, and effectiveness in managing the complexities of scRNA-seq data.

Figure 2 presents a detailed comparison of scMSCF with other leading clustering methods across various single-cell RNA sequencing datasets. Additional results for Variation of Information (VI) and Davies-Bouldin Index (DBI) are available in supplementary document SF1 and SF2. Overall, scMSCF demonstrates outstanding clustering performance across eight diverse datasets. It achieved the highest scores in the ARI, NMI, and ACC eight, six, and seven times, respectively, showcasing its stability and robustness with varied datasets. On average, scMSCF outperforms the second-best methods (such as Seurat or CellVGAE) by approximately 8–10%, indicating its strong capability to capture complex data structures and classifications. Notably, scMSCF also surpasses deep learning-based clustering algorithms like scSemiAAE, scMAE, scDSC, and scDeepCluster[47]. Across the eight tested datasets, scMSCF achieved ARI, NMI, and ACC scores that were 10–15% higher on average than these methods, underscoring its strengths in deep feature recognition and pattern classification. Overall, scMSCF leads in clustering performance across most datasets, particularly excelling in complex and highly heterogeneous datasets. Compared to traditional methods and other deep learning approaches, scMSCF showcases a superior ability to capture and classify single-cell data, further validating its potential and practicality in single-cell RNA sequencing data analysis.

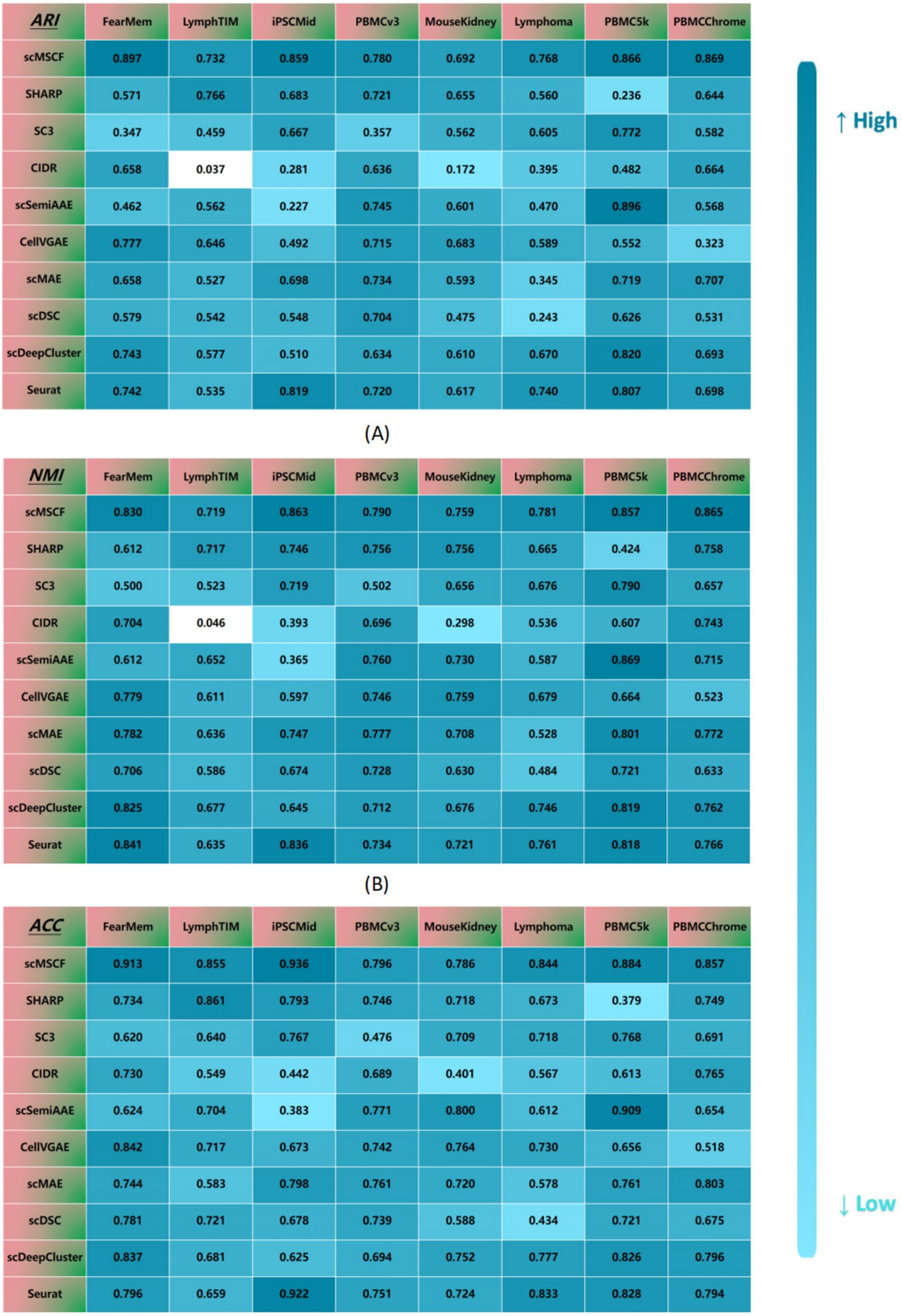
Figure 3 displays t-SNE 2D visualizations of clustering results from various algorithms on the 5k Healthy Donor PBMCs dataset, clearly delineating performance differences between scMSCF and competing methods. In comparison to traditional methods such as scDeepCluster and SC3, scMSCF demonstrates significant advantages in terms of clustering precision and integrity. While methods like SHARP and scSemiAAE achieve relatively good separation in some clusters, they still encounter issues such as irregular cluster shapes, overlap between clusters, and limited separation. In contrast, scMSCF, leveraging the deep feature extraction and self-attention mechanisms of the Transformer model, produces accurate and stable clusters in highly heterogeneous datasets, minimizing intra-cluster mixing and inter-cluster overlap. Furthermore, the Transformer model's ability to adapt to varying confidence levels during training enhances its capability to balance the quality and diversity of the training set, which in turn improves clustering stability and generalization. Results on additional datasets are provided in Supplementary document SF3–SF9.

To directly demonstrate the clustering efficacy, Fig. 4. employs a Sankey diagram to illustrate the flow of data points between actual labels and the results generated by the clustering algorithms. The diagram clearly displays the correspondence between true cell type labels and the clustering labels generated by scMSCF. Despite some dispersion for cell types like Endo and Ependy across clusters, scMSCF accurately assigns most cells to the optimal clusters. This performance is driven by its multi-step dimensionality reduction strategy and confidence ratio filtering mechanism. scMSCF first leverages Transformer-based deep feature extraction, using the self-attention mechanism to capture complex intercellular relationships, thus preserving crucial information on cell types during dimensionality reduction. Additionally, by selecting high-confidence cells as part of the training set, it achieves a balance between sample diversity and quality, allowing the model to learn the data's true structure more robustly. The application of a weighted strategy further enhances the differentiation of various cell types, enabling the model to form distinct clusters even with heterogeneous data. Overall, Fig. 4. effectively emphasizes the strengths of scMSCF in the analysis of single-cell RNA sequencing data. It not only achieves high accuracy and consistency but also excels at handling complex cell populations, establishing scMSCF as a powerful tool for clustering single-cell data.

To complement the evaluations on real-world datasets, we also tested scMSCF on simulated datasets to assess its scalability and runtime efficiency. The detailed setup and results of these simulations are provided in Supplementary Document SN5.

Effectiveness of transformer in enhancing clustering performance

To assess the impact of the scMSCF model on clustering performance within single-cell RNA sequencing data, we performed a comprehensive analysis across various datasets, focusing on ARI changes before and after implementing scMSCF. Figure 5 illustrates the comparison between initial ARI values and those obtained after applying scMSCF, underscoring the model's effectiveness in enhancing clustering outcomes. Overall, scMSCF demonstrates significant performance enhancements on most datasets. For instance, in the FearMem and PBMCChrome datasets, ARI values increased to 0.897 and 0.869 after scMSCF processing, up from initial values of 0.860 and 0.850, representing increases of approximately 4.3% and 2.2%. These results indicate that scMSCF, via its Transformer module, adeptly captures complex intercellular relationships, significantly improving clustering accuracy. While the extent of improvement varies across datasets, the overall trend is positive. Even in datasets with smaller gains, such as Lymphoma and MouseKidney, scMSCF raised ARI values from 0.759 to 0.683 to 0.768 and 0.692, respectively, demonstrating incremental yet meaningful enhancements. In conclusion, scMSCF consistently improves clustering performance across most datasets, highlighting the significant benefits



ARI	FearMem	LymphTIM	iPSCMid	PBMCv3	MouseKidney	Lymphoma	PBMC5k	PBMCChrome
scMSCF	0.897	0.732	0.859	0.780	0.692	0.768	0.866	0.869
SHARP	0.571	0.766	0.683	0.721	0.655	0.560	0.236	0.644
SC3	0.347	0.459	0.667	0.357	0.562	0.605	0.772	0.582
CIDR	0.658	0.037	0.281	0.636	0.172	0.395	0.482	0.664
scSemiAAE	0.462	0.562	0.227	0.745	0.601	0.470	0.896	0.568
CellVGAE	0.777	0.646	0.492	0.715	0.683	0.589	0.552	0.323
scMAE	0.658	0.527	0.698	0.734	0.593	0.345	0.719	0.707
scDSC	0.579	0.542	0.548	0.704	0.475	0.243	0.626	0.531
scDeepCluster	0.743	0.577	0.510	0.634	0.610	0.670	0.820	0.693
Seurat	0.742	0.535	0.819	0.720	0.617	0.740	0.807	0.698

(A)

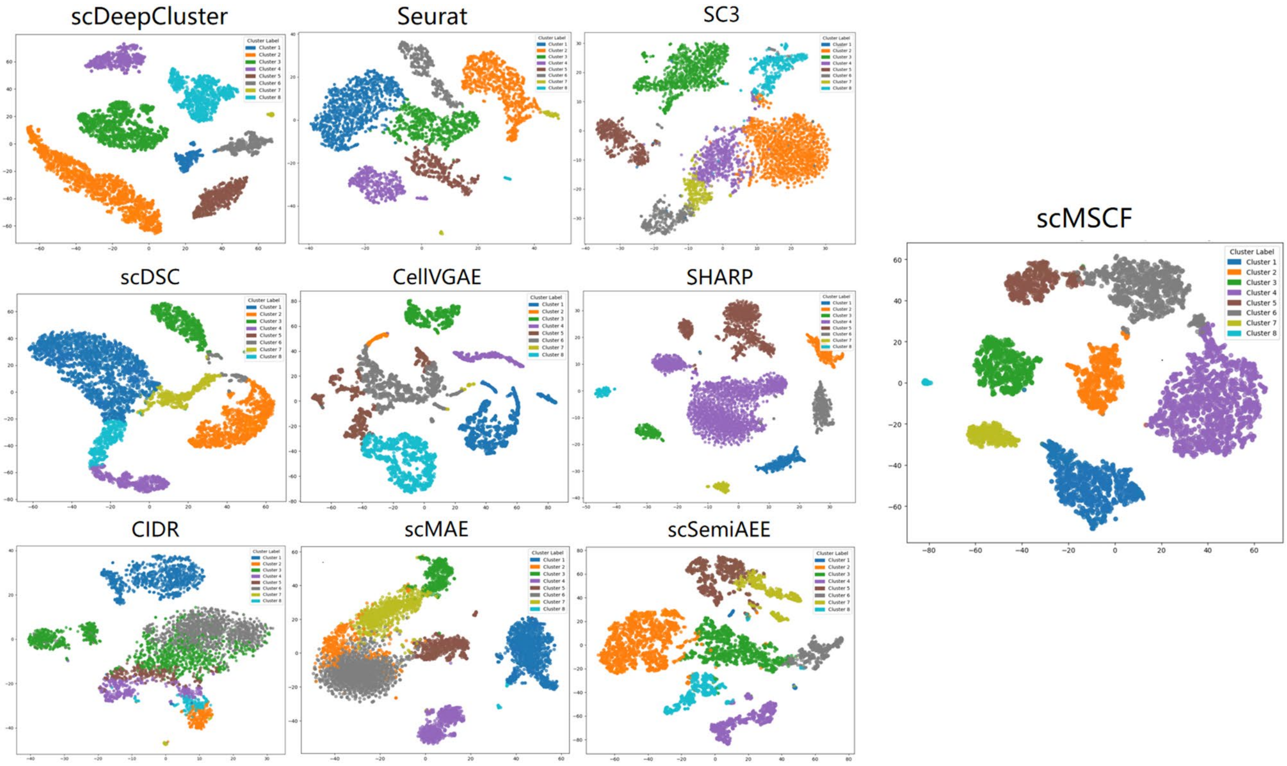
NMI	FearMem	LymphTIM	iPSCMid	PBMCv3	MouseKidney	Lymphoma	PBMC5k	PBMCChrome
scMSCF	0.830	0.719	0.863	0.790	0.759	0.781	0.857	0.865
SHARP	0.612	0.717	0.746	0.756	0.756	0.665	0.424	0.758
SC3	0.500	0.523	0.719	0.502	0.656	0.676	0.790	0.657
CIDR	0.704	0.046	0.393	0.696	0.298	0.536	0.607	0.743
scSemiAAE	0.612	0.652	0.365	0.760	0.730	0.587	0.869	0.715
CellVGAE	0.779	0.611	0.597	0.746	0.759	0.679	0.664	0.523
scMAE	0.782	0.636	0.747	0.777	0.708	0.528	0.801	0.772
scDSC	0.706	0.586	0.674	0.728	0.630	0.484	0.721	0.633
scDeepCluster	0.825	0.677	0.645	0.712	0.676	0.746	0.819	0.762
Seurat	0.841	0.635	0.836	0.734	0.721	0.761	0.818	0.766

(B)

ACC	FearMem	LymphTIM	iPSCMid	PBMCv3	MouseKidney	Lymphoma	PBMC5k	PBMCChrome
scMSCF	0.913	0.855	0.936	0.796	0.786	0.844	0.884	0.857
SHARP	0.734	0.861	0.793	0.746	0.718	0.673	0.379	0.749
SC3	0.620	0.640	0.767	0.476	0.709	0.718	0.768	0.691
CIDR	0.730	0.549	0.442	0.689	0.401	0.567	0.613	0.765
scSemiAAE	0.624	0.704	0.383	0.771	0.800	0.612	0.909	0.654
CellVGAE	0.842	0.717	0.673	0.742	0.764	0.730	0.656	0.518
scMAE	0.744	0.583	0.798	0.761	0.720	0.578	0.761	0.803
scDSC	0.781	0.721	0.678	0.739	0.588	0.434	0.721	0.675
scDeepCluster	0.837	0.681	0.625	0.694	0.752	0.777	0.826	0.796
Seurat	0.796	0.659	0.922	0.751	0.724	0.833	0.828	0.794

(C)

Fig. 2. Comparison of Clustering Performance between scMSCF and Nine Other Methods. This figure shows comparison plots for three metrics: ARI (**A**), NMI (**B**), and ACC (**C**). The remaining metrics, VI and DBI, are available in the supplementary materials. Note that CIDR[46] produced outlier values below 0.1 on some datasets; to improve visualization, the corresponding cells in the table are set to white.



scDeepCluster

Seurat

SC3

scMSCF

scDSC

CellVGAE

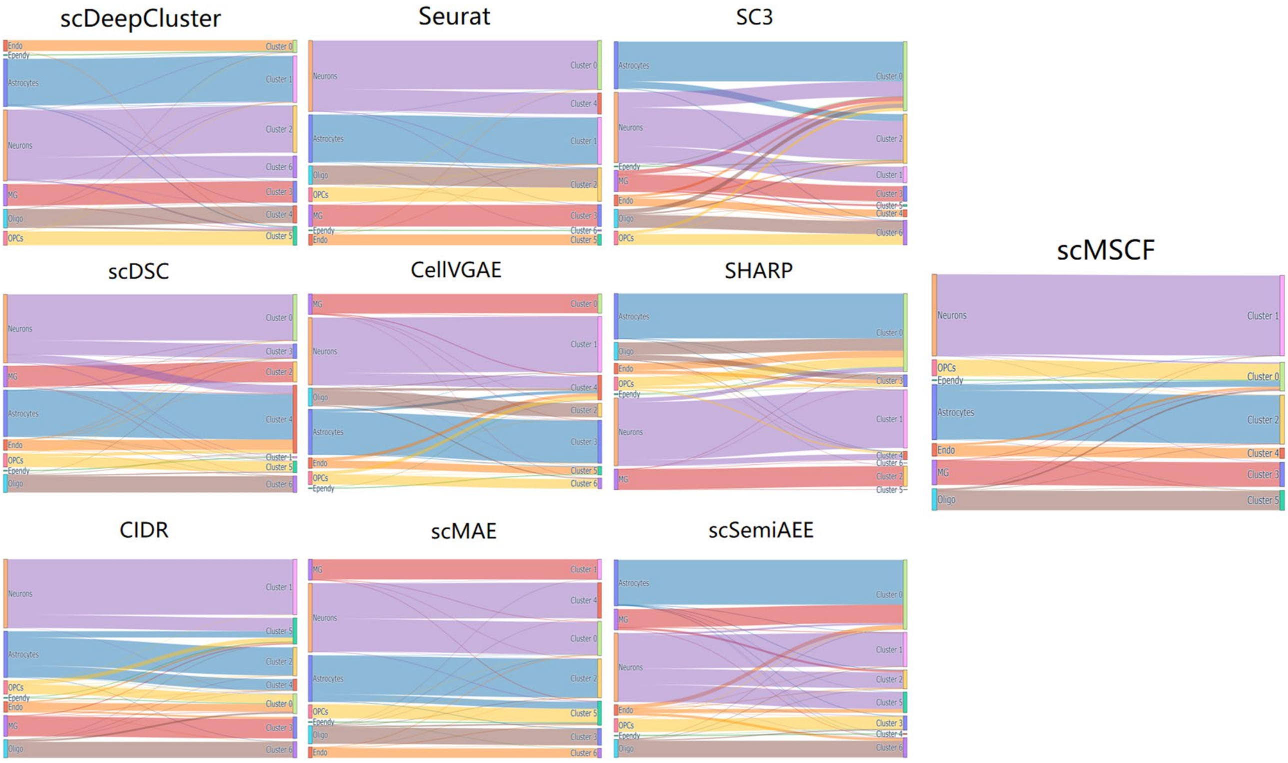
SHARP

CIDR

scMAE

scSemiAEE

Fig. 3. Visualization of clustering results for scMSCF and nine competing methods on the PBMC5k dataset.



scDeepCluster

Seurat

SC3

scMSCF

scDSC

CellVGAE

SHARP

CIDR

scMAE

scSemiAEE

Fig. 4. The Sankey diagram comparing scMSCF and nine other methods on the fear memory in the basolateral amygdala dataset.

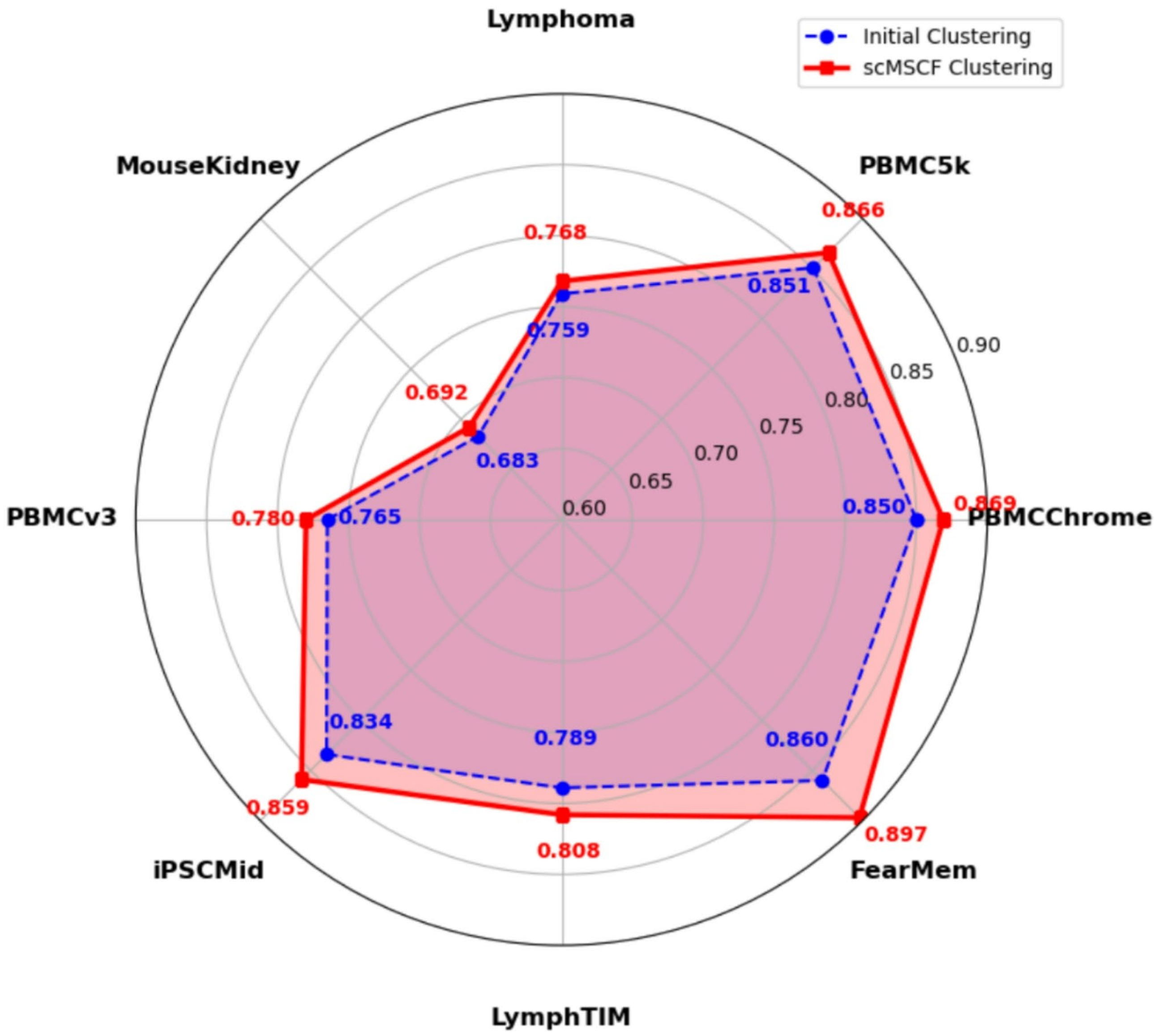


Fig. 5. ARI performance comparison between initial clustering and scMSCF clustering enhanced by the basolateral model.

of integrating the Transformer model into single-cell RNA sequencing data analysis. Despite some variation in performance gains, scMSCF achieves significant overall progress in clustering accuracy and stability.

Differential expression analysis

Differential expression analysis is essential in single-cell RNA sequencing data clustering, as it reveals gene expression differences that help identify and distinguish cell types. This analysis offers critical insights into cellular heterogeneity and provides robust support for accurate cell classification. In this study, we evaluate the marker gene identification performance of scMSCF against Seurat, SHARP, and CellVAGE, all of which have demonstrated robust ARI scores in single-cell RNA sequencing data. Figure 6 illustrates the expression levels and distribution patterns of the top five significantly differentially expressed genes within each cell cluster. As shown, scMSCF more effectively identifies key marker genes linked to cell subtypes.

To further substantiate scMSCF's efficacy in marker gene identification, Fig. 7 presents the average overlap rates between identified and true marker genes for scMSCF, Seurat, SHARP, and CellVAGE across three distinct datasets. Results show that scMSCF consistently achieves a higher overlap rate across all cell groups, indicating superior accuracy and consistency in cell type recognition. This performance highlights scMSCF's strengths in accurately capturing differentially expressed genes and demonstrates its high stability and precision when handling complex multicellular data, affirming scMSCF's potential as a powerful tool for analyzing complex single-cell data. Additionally, Supplementary File SF10 provides cluster-specific overlap rates across three datasets for the scMSCF algorithm.

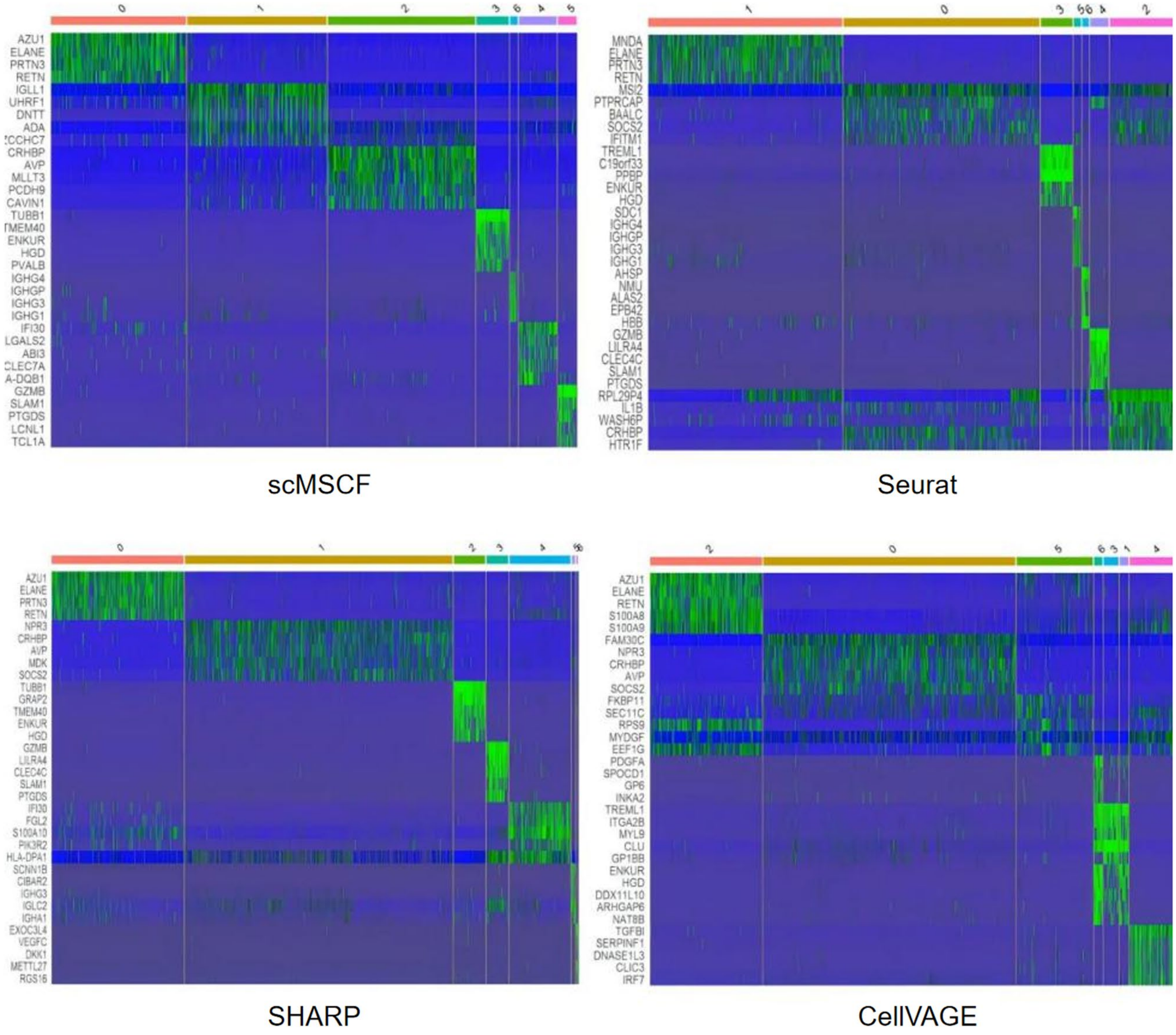


Fig. 6. Heatmap illustrating differences in marker gene identification among scMSCF, Seurat, SHARP, and CellVAGE on the mouse skin cells dataset.

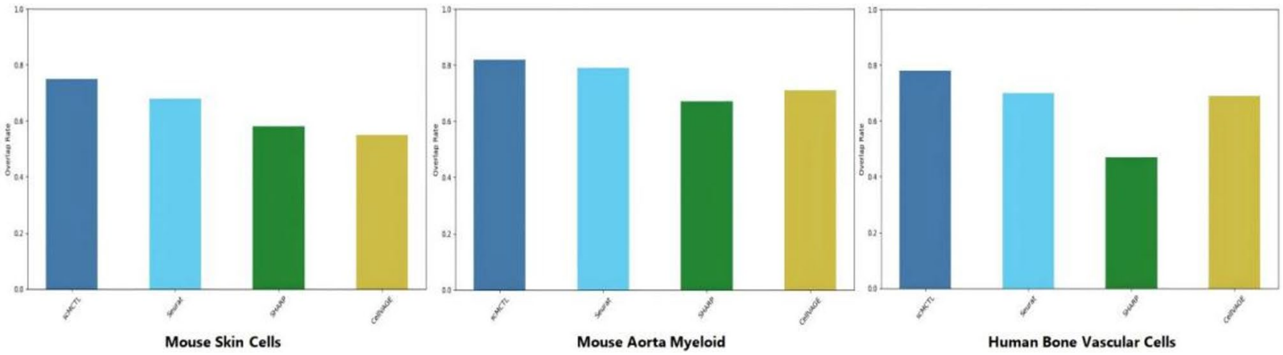


Fig. 7. Average overlap rate of identified marker genes with true marker genes for scMSCF, Seurat, SHARP, and CellVAGE across mouse skin cells, mouse aorta myeloid, and human bone vascular cells datasets.

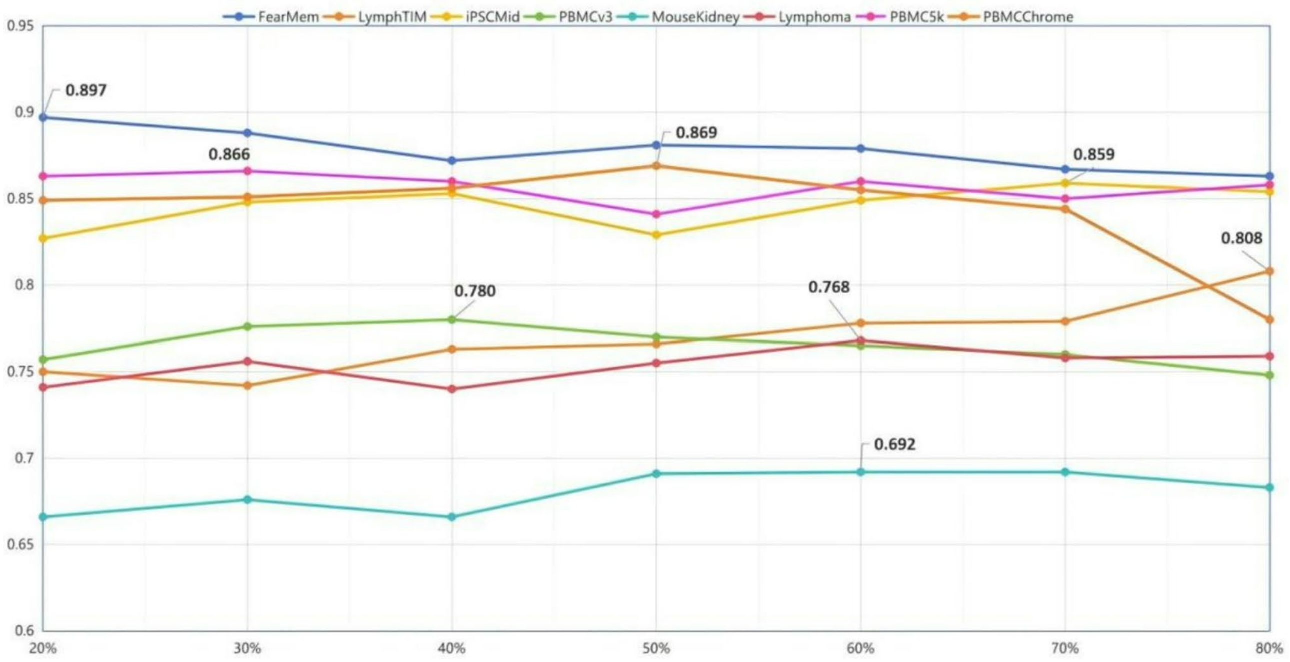


Fig. 8. Trends in ARI across different dataset sizes for scMSCF as the confidence ratio varies from 20–80%. The x-axis represents the confidence ratio percentage, and the y-axis shows the ARI (Adjusted Rand Index), which evaluates clustering accuracy. The datasets included are: FearMem (6361 cells), LymphTIM (615 cells), iPSCMid (767 cells), PBMCv3 (4352 cells), MouseKidney (1385 cells), Lymphoma (2913 cells), PBMC5k (5025 cells), and PBMCChrome (3363 cells). This figure highlights how different confidence ratios affect clustering performance across datasets of varying sizes, providing insights into the optimal confidence levels for different data scales.

To elucidate the dynamic evolution of cell states and further validate the scMSCF algorithm's capabilities in analyzing complex biological processes, pseudotime trajectory analysis was conducted. This method captures dynamic transitions between cell states and their temporal progression, offering a comprehensive framework to assess the algorithm's effectiveness in modeling cellular dynamics. Detailed methodologies and results are available in Supplementary Document SN6.

To provide deeper biological insight into the marker genes identified by scMSCF, we performed additional functional annotation and pathway enrichment analyses. The results indicate that these marker genes significantly participate in immune-related biological processes and signaling pathways, emphasizing their crucial roles in immune response regulation, inflammatory reactions, and intercellular communication. Detailed methodologies and comprehensive enrichment results for these analyses are provided in Supplementary Document SN7.

Ablation study

This ablation study accesses the impact of the confidence threshold on model performance. In particular, the scMSCF method selects subsets of high-confidence cells—determined by their proportion in the dataset, known as the confidence ratio—and utilizes these subsets as the entire training set for deep learning via the Transformer model. Choosing the right proportion of high-confidence cells is crucial for effective model training. Setting a high threshold, such as 80%, may result in a training set overly concentrated with high-confidence cells, thereby reducing sample diversity and increasing the risk of overfitting. This can cause the model to miss valuable information from low-confidence samples, thereby diminishing its generalization ability. Conversely, a low confidence ratio, such as 20%, might incorporate too many low-confidence samples into the training set, introducing noise that undermines model stability and accuracy. Low-confidence samples are typically associated with higher uncertainty, making it harder for the model to learn meaningful features, which in turn affects clustering accuracy. Balancing the proportion of high-confidence cells is essential to optimize the training set's quality and diversity, thus enhancing overall model performance. In this experiment, we evaluated datasets with varying sizes using confidence ratios ranging from 20 to 80%. Extreme values of 10% and 90% were excluded due to their impracticality for model training. A 10% threshold would yield a training set dominated by low-confidence samples, leading to unstable training, while a 90% threshold would excessively reduce training diversity, impairing model generalization.

By evaluating thresholds ranging from 20 to 80%, our goal was to identify the optimal confidence ratio that ensures a balance between training set quality and diversity. Based on the results in Fig. 8, the choice of confidence ratio has a significant impact on scMSCF's clustering performance. We suggest the following ranges: for datasets with fewer than 3000 cells, a confidence ratio of 80–60%; for datasets with 3000 to 4500 cells, a range of 50–40%; and for datasets with more than 4500 cells, a range of 30–20%.

In addition to the confidence threshold analysis presented above, we further investigated the sensitivity of other key hyperparameters, including the number of Transformer layers, feature dimension (d_model), and learning rate. These experiments, detailed in Supplementary Document SN8, provide additional insights into how different parameter settings affect clustering performance and offer practical guidance for robust hyperparameter tuning.

Discussion

In this study, we proposed scMSCF, an innovative multi-step clustering framework designed to address the challenges of high-dimensionality, sparsity, and noise commonly associated with single-cell RNA sequencing (scRNA-seq) data analysis. By integrating multi-dimensional reduction strategies, weighted ensemble clustering methods, and Transformer-based deep learning techniques, scMSCF demonstrated exceptional clustering performance, robustness, and scalability. Evaluations on multiple real and simulated datasets showed that scMSCF significantly outperformed mainstream methods in key metrics such as Adjusted Rand Index (ARI) and Normalized Mutual Information (NMI). Notably, scMSCF introduced a confidence-based training set selection mechanism, which identifies high-confidence cells through a voting strategy within the weighted ensemble clustering process. This innovation greatly improved the reliability and diversity of the training data, providing a robust foundation for the Transformer model's deep feature learning. Consequently, scMSCF achieved superior performance in identifying rare cell types and resolving heterogeneous cell populations, particularly in complex biological scenarios.

In addition to its clustering accuracy, scMSCF exhibited excellent computational efficiency and scalability. Across simulated datasets of varying sizes, scMSCF showed an approximately linear increase in runtime with data size, indicating its suitability for large-scale data analysis. Furthermore, scMSCF excelled in pseudotime trajectory analysis, further highlighting its ability to extract meaningful biological insights. For example, in Parkinson's disease-related iPSC midbrain neuron datasets, scMSCF not only accurately revealed dynamic cell state transitions over time but also aligned pseudotime results closely with actual time points, offering a clear depiction of cell transitions from early developmental stages to maturity. These results underscore scMSCF's potential as a powerful tool for analyzing complex single-cell datasets while supporting studies of cellular differentiation and disease progression.

To further substantiate the computational practicality of scMSCF, we conducted runtime and GPU memory profiling under two distinct high-performance computing platforms. As reported in Supplementary Document SN9, scMSCF was tested on both small- and medium-scale datasets (iPSC Midbrain and PBMC5k, respectively), and results showed that training and inference completed in under 10 min on typical modern GPUs, with memory usage staying well below 10 GB. These findings not only confirm scMSCF's runtime scalability but also its deployability across diverse computing environments, reinforcing its suitability for real-world large-scale applications.

Despite its significant advantages, scMSCF has certain limitations. First, scMSCF is sensitive to hyperparameter settings, requiring careful tuning to achieve optimal performance, especially regarding confidence thresholds and Transformer model parameters. Second, its effectiveness relies on the consistency and quality of initial clustering results, as these directly influence the selection of training cells and subsequent model performance. These limitations suggest areas for future improvement, particularly in developing methods to enhance robustness to parameter variation and initial clustering quality, further broadening the applicability and ease-of-use of scMSCF in diverse practical scenarios.

Data Availability

The six scRNA-seq datasets analyzed in this study are available in the Gene Expression Omnibus (GEO) repository under the accession numbers GSE246147(FearMem), GSE154763(LymphTIM), GSE247600(iPSCMid).The PBMCv3, MouseKidney, Lymphoma, PBMC5k, and PBMCChrome datasets analyzed in this study can be accessed from the 10X Genomics repository at the following links: https://www.10xgenomics.com/datasets/peripheral-blood-mononuclear-cells-pbm-cs-from-a-healthy-donor-manual-channel-1–3-1-standard-3-1-0(PBMCv3), https://www.10xgenomics.com/datasets/1k-mouse-kidney-nuclei-isolated-with-chromium-nuclei-isolation-kit-3-1-standard(MouseKidney), https://www.10xgenomics.com/datasets/hodgkins-lymphoma-dissociated-tumor-targeted-compare-immunology-panel-3-1-standard-4-0-0(Lymphoma), https://www.10xgenomics.com/datasets/5-k-peripheral-blood-mononuclear-cells-pbm-cs-from-a-healthy-donor-v-3-chemistry-3-1-standard-3-0-2(PBMC5k), and https://www.10xgenomics.com/datasets/peripheral-blood-mononuclear-cells-pbm-cs-from-a-healthy-donor-chromium-connect-channel-5-3-1-standard-3-1-0(PBMCChrome). The dataset used in the differential expression analysis of this study can be obtained from the Array Express repository with login number E-MTAB-7417(Mouse), E-MTAB-10,432(Human) and E-MTAB-10,746(Myeloid).

Received: 11 January 2025; Accepted: 21 May 2025
Published online: 27 May 2025

References

1. Lähnemann, D. et al. Eleven grand challenges in single-cell data science. *Genome Biol.* **21** (1). https://doi.org/10.1186/s13059-020-1926-6 (2020). 31, DOI.
2. Klein, A. M. et al. Droplet barcoding for single-cell transcriptomics applied to embryonic stem cells. *Cell* **161** (5), 1187–1201. https://doi.org/10.1016/j.cell.2015.04.044 (2015).
3. Macosko, E. Z. et al. Highly parallel genome-wide expression profiling of individual cells using nanoliter droplets. *Cell* **161** (5), 1202–1214. https://doi.org/10.1016/j.cell.2015.05.002 (2015).

4. Kiselev, V. Y., Andrews, T. S. & Hemberg, M. Challenges in unsupervised clustering of single-cell RNA-seq data. *Nat. Rev. Genet.* **20** (5), 273–282. https://doi.org/10.1038/s41576-018-0088-9 (2019).
5. Levine, J. H. et al. Data-driven phenotypic dissection of AML reveals progenitor-like cells that correlate with prognosis. *Cell* **162** (1), 184–197. https://doi.org/10.1016/j.cell.2015.05.047 (2015).
6. Haghverdi, L., Lun, A. T. L., Morgan, M. D. & Marioni, J. C. Batch effects in single-cell RNA-sequencing data are corrected by matching mutual nearest neighbors. *Nat. Biotechnol.* **36** (5), 421–427. https://doi.org/10.1038/nbt.4091 (2018).
7. McLachlan, G. J., Bean, R. W. & Peel, D. A mixture model-based approach to the clustering of microarray expression data. *Bioinformatics* **18** (3), 413–422. https://doi.org/10.1093/bioinformatics/18.3.413 (2002).
8. Von Luxburg, U. A tutorial on spectral clustering. *Stat. Comput.* **17**, 395–416. https://doi.org/10.1007/s11222-007-9033-z (2007).
9. Butler, A. et al. Integrating single-cell transcriptomic data across different conditions, technologies, and species. *Nat. Biotechnol.* **36** (5), 411–420. https://doi.org/10.1038/nbt.4096 (2018).
10. Stuart, T. et al. Comprehensive integration of single-cell data. *Cell* **177** (7), 1888–1902e21. https://doi.org/10.1016/j.cell.2019.05.031 (2019).
11. Xu, C. & Su, Z. Identification of cell types from single-cell transcriptomes using a novel clustering method. *Bioinformatics* **31** (12), 1974–1980. https://doi.org/10.1093/bioinformatics/btv088 (2015).
12. van der Maaten, L. & Hinton, G. Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9, 2579–2605. (2008). Available at: http://www.jmlr.org/papers/v9/vandermaaten08a.html
13. Hahsler, M., Piekenbrock, M., Doran, D. & dbscan Fast density-based clustering with R. *J. Stat. Softw.* **91** (1), 1–30. https://doi.org/10.18637/jss.v091.i01 (2019).
14. Wang, B. et al. SIMLR: a tool for large-scale single-cell RNA-seq data analysis by multi-kernel learning. *Genome Biol.* **18**, 86. https://doi.org/10.1101/118901 (2017).
15. Kiselev, V. Y. et al. SC3: consensus clustering of single-cell RNA-seq data. *Nat. Methods.* **14** (5), 483–486. https://doi.org/10.1038/nmeth.4236 (2017).
16. Wan, S., Kim, J. & Won, K. J. SHARP: hyperfast and accurate processing of single-cell RNA-seq data via ensemble random projection. *Genome Res.* **33**, 1–14. https://doi.org/10.1101/gr.254557.119 (2020).
17. Eraslan, G. et al. Single-cell RNA-seq denoising using a deep count autoencoder. *Nat. Commun.* **10** (1). https://doi.org/10.1038/s41467-018-07931-2 (2019). 390, DOI.
18. Raza, K. Machine learning in single-cell RNA-seq data analysis. *Springer* https://doi.org/10.1007/978-981-97-6703-8 (2024).
19. Brendel, M. et al. Application of deep learning on Single-cell RNA sequencing data analysis: A review. *Genom. Proteom. Bioinform.* **20** (5), 814–835. https://doi.org/10.1016/j.gpb.2022.11.011 (2022).
20. Gan, Y., Huang, X., Zou, G., Zhou, S. & Guan, J. Deep structural clustering for single-cell RNA-seq data jointly through autoencoder and graph neural network. *Brief. Bioinform.* **23** (2), 1–13. https://doi.org/10.1093/bib/bbac018 (2022).
21. Fang, Z., Zheng, R. & Li, M. ScMAE: a masked autoencoder for single-cell RNA-seq clustering. *Bioinformatics* **40** (1). https://doi.org/10.1093/bioinformatics/btae020 (2024).
22. He, K. et al. Masked Autoencoders Are Scalable Vision Learners. *arXiv preprint*, arXiv:2111.06377, (2021). https://doi.org/10.48550/arXiv.2111.06377
23. Buterez, D., Bica, I., Tariq, I., Andrés-Terré, H. & Liò, P. CellVGAE: an unsupervised scRNA-seq analysis workflow with graph attention networks. *Bioinformatics* **38** (5), 1277–1286. https://doi.org/10.1093/bioinformatics/btab804 (2022).
24. Wang, J. et al. ScGNN is a novel graph neural network framework for single-cell RNA-Seq analyses. *Nat. Commun.* **12**, 1882. https://doi.org/10.1038/s41467-021-22197-x (2021).
25. Grønbæk, C. H. et al. ScVAE: variational auto-encoders for single-cell gene expression data. *Bioinformatics* **36** (16), 4415–4422. https://doi.org/10.1093/bioinformatics/btaa293 (2020).
26. Li, X. et al. Deep learning enables accurate clustering with batch effect removal in single-cell RNA-seq analysis. *Nat. Commun.* **11**, 2338. https://doi.org/10.1038/s41467-020-15851-3 (2020).
27. P Kingma, D., J Rezende, D., Mohamed, S. & Welling, M. Semi-Supervised learning with deep generative models. *ArXiv Preprint.* https://doi.org/10.48550/arXiv.1406.5298 (2014). arXiv:1406.5298.
28. Wang, Z., Wang, H., Zhao, J. & Zheng, C. ScSemiAAE: a semi-supervised clustering model for single-cell RNA-seq data. *BMC Bioinform.* **24**, 217. https://doi.org/10.1186/s12859-023-05339-4 (2023).
29. Risso, D. et al. A general and flexible method for signal extraction from single-cell RNA-seq data. *Nat. Commun.* **9** (1), 284. https://doi.org/10.1038/s41467-017-02554-5 (2018).
30. Goodfellow, I. et al. Generative adversarial networks. *Commun. ACM.* **63** (11), 139–144. https://doi.org/10.1145/3422622 (2020).
31. Jiang, H., Yang, L., Zou, Q., Qiu, Y. & scTPC A novel semi-supervised deep clustering model for scRNA-seq data. *Bioinformatics* **40** (5). https://doi.org/10.1093/bioinformatics/btae293 (2024).
32. Lin, T. Y., Goyal, P., Girshick, R., He, K. & Dollár, P. Focal Loss for Dense Object Detection. *arXiv preprint*, arXiv:1708.02002, (2017). https://doi.org/10.48550/arXiv.1708.02002
33. Zhao, J. P., Hou, T. S., Su, Y. S. & Zheng, C. H. ScSSA: A clustering method for single-cell RNA-seq data based on semi-supervised autoencoder. *Methods* **208**, 66–74. https://doi.org/10.1016/j.ymeth.2022.10.006 (2022).
34. Deng, Y., Bao, F., Dai, Q., Wu, L. F. & Altschuler, S. J. Scalable analysis of cell type composition from single-cell transcriptomics using deep recurrent learning. *Nat. Methods.* **16** (4), 311–314. https://doi.org/10.1038/s41592-019-0353-7 (2019).
35. Yu, B. et al. ScGMAI: a Gaussian mixture model for clustering single-cell RNA-seq data based on deep autoencoder. *Brief. Bioinform.* **22** (4). https://doi.org/10.1093/bib/bbaa316 (2020).
36. Zhao, J., Shang, C., Li, S., Xin, L. & Yu, P. L. H. Choosing the number of factors in factor analysis with incomplete data via a hierarchical Bayesian information criterion. *arXiv preprint*, arXiv:2204.09086, (2022). https://doi.org/10.48550/arXiv.2204.09086
37. Liu, Q., Liang, Y., Wang, D. & Li, J. L. F. S. C. A linear fast semi-supervised clustering algorithm that integrates reference-bulk and single-cell transcriptomes. *Front. Genet.* **13**, 1068075. https://doi.org/10.3389/fgene.2022.1068075 (2022).
38. Tian, L. P., Liu, L. & Wu, F. X. Matrix decomposition methods in bioinformatics. *Curr. Bioinform.* **8** (2), 259–266. https://doi.org/10.2174/1574893611308020014 (2013).
39. Wang, M., Fu, W., Hao, S., Tao, D. & Wu, X. Scalable Semi-Supervised learning by efficient anchor graph regularization. *IEEE Trans. Knowl. Data Eng.* **28** (7), 1864–1877. https://doi.org/10.1109/TKDE.2016.2535367 (2016).
40. Wolf, T. et al. Transformers: State-of-the-Art natural Language processing. *Proc. 2020 Conf. Empir. Methods Nat. Lang. Processing: Syst. Demonstrations.* **38-45** https://doi.org/10.18653/v1/2020.emnlp-demos.6 (2020).
41. Hubert, L., Arabie, P. & Comparing partitions *J. Classif.*, **2** (2), 193–218. DOI: https://doi.org/10.1007/BF01908075 (1985).
42. Petegrosso, R., Li, Z. & Kuang, R. Machine learning and statistical methods for clustering single-cell RNA-sequencing data. *Brief. Bioinform.* **21** (4), 1209–1223. https://doi.org/10.1093/bib/bbz063 (2020).
43. Fränti, P. & Sieranoja, S. Clustering accuracy. *Appl. Comput. Intell.* **4** (1), 24–44. https://doi.org/10.3934/aci.2024003 (2024).
44. Meilă, M. Comparing clusterings—an information based distance. *J. Multivar. Anal.* **98** (5), 873–895. https://doi.org/10.1016/j.jmva.2006.11.013 (2007).
45. Xiao, J., Lu, J. & Li, X. Davies Bouldin index based hierarchical initialization K-means. *Intell. Data Anal.* **21** (6), 1327–1338. https://doi.org/10.3233/IDA-163129 (2017).
46. Lin, P., Troup, M., Ho, J. W. & CIDR Ultrafast and accurate clustering through imputation for single-cell RNA-seq data. *Genome Biol.* **18** (1), 59. https://doi.org/10.1186/s13059-017-1188-0 (2017).

47. Tian, T., Zhang, H., Wang, Q. & Cai, Z. Clustering single-cell RNA-seq data with a model-based deep learning approach. *Nat. Mach. Intell.* **1** (3), 191–198. https://doi.org/10.1038/s42256-019-0037-0 (2019).

Acknowledgements

The work described in this paper is supported by the Foundation of Science and Technology Department of Jilin Province under Grant No. YDZJ202201ZYTS531.

Author contributions

S.J. wrote the main manuscript text, prepared figures and dataset table and performed the data analysis. C.W. and Q.S. developed the methodology and designed the study. S.J. and Z.Z. conducted the experiments. All authors reviewed the manuscript.

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1038/s41598-025-03603-6.

Correspondence and requests for materials should be addressed to C.W. or Q.S.