BIG DATA MINING AND ANALYTICS
ISSN 2096-0654 10/25 pp1169–1186
DOI: 10.26599/BDMA.2024.9020034
Volume 7, Number 4, December 2024

Transformer-Based Single-Cell Language Model: A Survey

Wei Lan, Guohang He, Mingyang Liu, Qingfeng Chen*, Junyue Cao*, and Wei Peng

Abstract: The transformers have achieved significant accomplishments in the natural language processing as its outstanding parallel processing capabilities and highly flexible attention mechanism. In addition, increasing studies based on transformers have been proposed to model single-cell data. In this review, we attempt to systematically summarize the single-cell language models and applications based on transformers. First, we provide a detailed introduction about the structures and principles of transformers. Then, we review the single-cell language models and large language models for single-cell data analysis. Moreover, we explore the datasets and applications of single-cell language models in downstream tasks, such as batch correction, cell clustering, cell type annotation, gene regulatory network inference, and perturbation response. Further, we discuss the challenges of single-cell language models and provide promising research directions. We hope this review will serve as an up-to-date reference for researchers who are interested in the direction of single-cell language models.

Key words: language model; transformers; deep learning; single-cell data

1 Introduction

Single-cell research has shown tremendous potential across a variety of fields, including genetics, immunology, and oncology. By utilizing single-cell RNA sequencing data for cluster analysis and the identification of cell subtypes, it is possible to accurately categorize cell populations and reveal crucial information about cell interactions and the structure of tissues[1]. Exploring the gene expression, gene function and gene-gene interaction at the single-cell level help to unveil the deep mechanisms of cellular heterogeneity within tissues[2, 3]. Single-cell research is critically important for understanding fundamental biological processes and provides significant insights for the diagnosis of diseases[4]. Single-cell data usually consist of large amounts of high-dimensional data which contain complex information. There is heterogeneity among single-cell data originating from the same tissue.

In the early stages, traditional machine learning methods, such as n-gram[5] and Hidden Markov Models (HMM)[6], were widely used for cell annotation and protein prediction. With the development of machine learning technology, more sophisticated algorithms are applied to single-cell research[7]. Subsequently, deep learning models, including Recurrent Neural Networks (RNN)[8] and Convolutional Neural Networks

• Wei Lan and Guohang He are with Guangxi Key Laboratory of Multimedia Communications and Network Technology, School of Computer, Electronic and Information, Guangxi University, Nanning 530004, China. E-mail: lanwei@gxu.edu.cn; 18775067872@163.com.
• Mingyang Liu and Qingfeng Chen are with School of Computer and Electronic Information, Guangxi University, Nanning 530004, China. E-mail: hitomil@foxmail.com; qingfeng@gxu.edu.cn.
• Junyue Cao is with College of Life Science and Technology, Guangxi University, Nanning 530004, China. E-mail: junyue.cao@gxu.edu.cn.
• Wei Peng is with Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, China. E-mail: weipeng1980@gmail.com.
* To whom correspondence should be addressed.
Manuscript received: 2024-02-23; revised: 2024-05-08; accepted: 2024-05-20

© The author(s) 2024. The articles published in this open access journal are distributed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/).

(CNN)[9], are used for the analysis of single-cell data. Currently, the transformers developed by Google has become the most popular language model[10]. The transformers can process an entire sentence at once during training and effectively captures long-distance dependencies within sequences through the self-attention mechanism[11]. This capability enables transformers to effectively explore various types of single-cell data. It leads to an increasing number of researchers applying transformer technology in the field of single-cell research[12].

This review will introduce the main modules of the transformers in Section 2. Then, we provide an overview and analysis of existing single-cell language models in Section 3 and showcase some downstream tasks accomplished by single-cell language models in Section 4. Final, we discuss the challenges and opportunities of transformers-based single-cell language models in Section 5. We hope to offer assistance to individuals who are interested in understanding single-cell language models.

2 Transformer

The transformers requires extensive training on numerous texts. It usually employs a self-supervised approach during training, enabling language models to perform classification and generation[13]. For instance, the transformers-based language models can automatically extract key information of text, generate new text and answer user queries in question-answering. These achievement is credited to the ability of transformers for learning long-term dependencies of language and allowing parallel training across multiple language units. This enhances the parallelism in processing sentences and capability to extract overall sequence correlations of transformers. The structure of transformers is depicted in Fig. 1.

The transformers have demonstrated excellent performance in both training tasks from scratch and pre-training tasks. Transformer-XL[14] introduces the recursive mechanism and positional encoding. It captures longer-term dependencies by learning beyond fixed-length dependencies while maintaining temporal continuity to address context fragmentation. Reformer[15] reduces attention calculation complexity and uses reversible residual layers instead of standard residual layers to achieve higher memory efficiency and alleviate pressure on computing resources. In addition, pre-training tasks can reduce dependence on annotated data, thus lowering the training cost of the transformers[16]. The Generative Pre-trained Transformer (GPT)[17] employs multiple layers of the transformers encoders and performs unsupervised

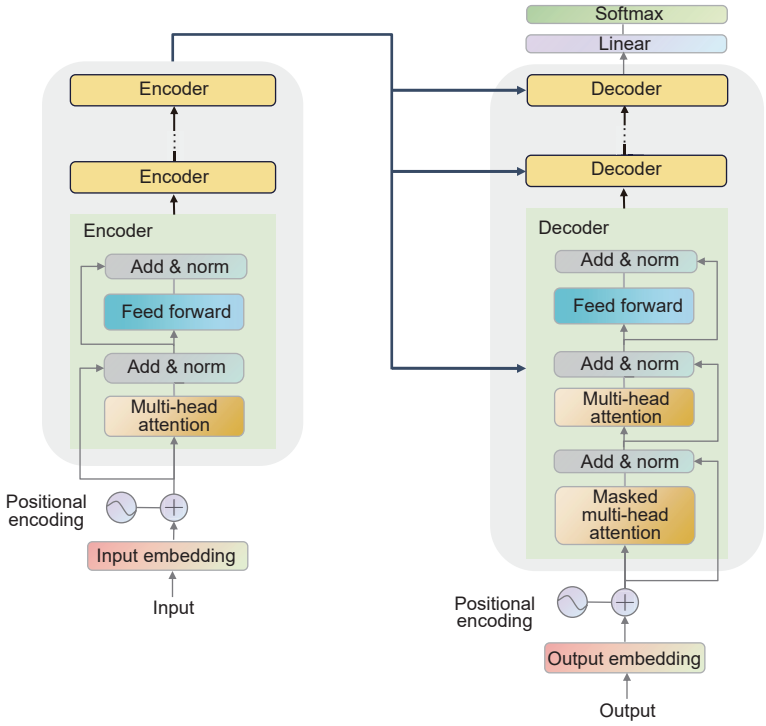


Fig. 1 Structure of transformers.

language modeling tasks during pre-training to learn semantic and syntactic knowledge from the text. The Bidirectional Encoder Representations from Transformers (BERT)[18] is a model that pre-trained on large datasets. It uses bi-directional transformers and mask mechanism to consider the context information from both the left and right sides of the input sequence simultaneously. Due to the success of these models, many models based on them have started to emerge. XLNet[19] is a pre-training model base on Transformer-XL that achieves bidirectional learning of context. It uses the self-regressive strategy to helps the model avoid the inconsistency issue in pre-training fine-tuning. RoBERTa[20] is a model based on BERT and achieves enhanced training performance by utilizing dynamic masking.

2.1 Encoder and decoder

The transformers is primarily composed of encoders and decoders, which uses residual connections and layer normalization. The layer normalization and residual connection are defined as follows:

$$\text{LayerNorm}\,(X+\text{MultiHead}\,(X)) \tag{1}$$

$$\text{LayerNorm}\,(X+\text{FFN}\,(X)) \tag{2}$$

where X in Formula (1) denotes the input embedding. It is processed through multi-head self-attention mechanism (MultiHead). After processing X, the result is added to the original X to obtain X in Formula (2). Then X in Formula (2) is processed through Feed-Forward Neural network (FFN). The layer normalization computes the mean and variance of each input sequence to provide more accurate training results[21]. The encoder gradually extracts semantic information from the input sequence and encodes it into a series of hidden vectors by stacking multiple identical layers. The decoder is responsible for transforming the hidden representations generated by the encoder into an output sequence. It adopts an autoregressive training approach. The decoder acquires information about the entire sequence of tokens during training, which would lead to a decrease in prediction accuracy. To address this issue, the decoder uses masked self-attention mechanism in the first layer. After obtaining vector information based on the masked self-attention mechanism, it needs to be combined with the hidden vectors provided by the encoder before entering the next layer. Then, the decoder gradually generates vectors of the sequence and transforms them into the final output sequence based on linear transformation and Softmax function.

2.2 Multi-head self-attention mechanism

The multi-head self-attention mechanism is comprised of multiple self-attention mechanisms. It can help the model to determine the important parameters during the training process. In addition, it adjusts the weights at different positions by calculating the correlations between each input position and other positions.

The self-attention mechanism is defined as follows:

$$\text{Attention}\,(Q,\,K,\,V)=\text{softmax}\,(\frac{Q\cdot K^{\mathrm{T}}}{\sqrt{d_k}})V \tag{3}$$

where d_k represents the dimensionality of the key vector, Q, K, and V are three matrices, and K^{T} represents the transpose of the K matrix. The dot product of Q and K^{T} denotes the similarity between the current word vector and other word vectors. After dividing this value by $\sqrt{d_k}$ and applying the softmax function, the coefficient of weight is obtained. The weight coefficient is then multiplied by V to ultimately obtain the attention value. The multi-head attention mechanism is defined as follows:

$$M\,(Q,\,K,\,V)=C\,(\text{head}_1,\,\text{head}_2,\,\ldots,\,\text{head}_h)W^O \tag{4}$$

$$\text{head}_i=\text{Attention}\,(QW_i^Q,\,KW_i^K,\,VW_i^V) \tag{5}$$

where W^O is a matrix containing the weights for each attention value，$M(\,)$ represents the multi-head attention function, $C(\,)$ denotes the concat function, head_i represents the self-attention mechanism module of i-th head, W^O contains the weights of each head_i, and W^Q, W^K, and W^V denote the weight matrices. Each input embedding vector is multiplied with them to obtain the corresponding matrices Q, K, and V. They are updated with each backward propagation during training. Each self-attention module has different W^Q, W^K, and W^V. The multi-head attention value is calculated by weighting each attention value with W^O.

2.3 Position encoding

The position encoding is obtained by adding positional information to the embedding vectors of input words in transformers. It is defined as follows:

$$\text{PE}_{(\text{pos},\,2i)}=\sin\,(\text{pos}/10000^{2i/d_{\text{model}}}) \tag{6}$$

$$\mathrm{PE}_{(\mathrm{pos},\,2i+1)} = \cos\left(\mathrm{pos}/10000^{2i/d_{\mathrm{model}}}\right) \tag{7}$$

where pos is the position index, i is the dimension index, and d_{model} is the size of the hidden layer. The sine and cosine values for each pos and i are calculated separately using the PE function, then they are merged into a position encoding vector. This ensures that the embedding vectors for each token not only contain semantic information, but also position information of input sequence. In addition, the relative position encoding is proposed in Ref. [22], to make transformers to better understand the positional information of the input sequence, thereby enhancing the performance and generalization capability of model.

2.4 Position-wise feed-forward networks

The position-wise feed-forward networks acts as a multi-layer perceptron, which is equivalent to use a linear layer in each encoder and decoder[23]. It is defined as follows:

$$\mathrm{FFN}(x) = \max(0,\, xW_1 + b_1)W_2 + b_2 \tag{8}$$

where W_1, b_1, W_2, and b_2 are parameters that can be learned during training. The FFN initially performs a linear operation on the input to increase its dimension and applies the ReLU activation function to learn more complex feature information. In final, the FFN reduces the dimension to the original dimension based on a linear operation to enhance the generalization capability of features.

3 Application of Transformer in Single-Cell

We categorize the application of single-cell data analysis based on transformers into single-cell language models and single-cell large language models depended on whether it uses pre-training or not. These models effectively analyze single-cell omics data by utilizing the unique feature representation of transformers.

3.1 Single-cell language model

This section introduces the current structural design and optimization of single-cell language models. These models are developed based on the transformers' framework. They have been utilized for analyzing various types of single-cell datasets, including single-cell transcriptomics, spatial transcriptomics, and epigenomics.

3.1.1 Single-cell language model based on single-cell transcriptomics

The transCluster[24] is a model based on transformers for analyzing scRNA-Seq data. It demonstrates that transformers can be used for scRNA-seq analysis. It utilizes Linear Discriminant Analysis (LDA)[25] to obtain input embeddings for the transformers. Then, CNN is employed to train the output of transformers for predicting cell types. In addition, scTransSort[26] is also a model that combines of transformers and CNN. It uses CNN to transfer the gene embeddings of each cell into multiple two-dimensional matrix blocks. Each matrix block represents a token and these tokens are trained through 12 layers of transformers. Finally, a linear classifier utilizes the output features of transformers to predict cell type. CIForm[27] is a model inspired by the application of transformers in Computer Vision (CV). It divides equally sized sub-vectors within the gene embedding module. These sub-vectors combined with positional embeddings are fed into the transformers for training. The output of the transformer is composed of sub-vectors and the average pooling layer is used to train the average values of these sub-vectors to obtain the final result. STGRNS[28] is an interpretable model base on transformers. It proposes a Gene Expression Motif (GEM) data processing technique to process scRNA-seq. The combination of GEM and transformers in STGRNS provides stonger interpretability. In contrast to STGRNS, T-GEM[29] enhances model interpretability by replacing the weights in the transformers with gene-related weights. It obtains attention values for different genes. Then, it utilizes these attention values for the classification task.

3.1.2 Single-cell language model based on single-cell spatial omics and epigenomics

The PROTRAIT[30] is a model based on transformers for analyzing scATAC-Seq data. It utilizes one-hot encoding to map input sequences into a latent space. When the sequence length is less than a predefined threshold, the one-hot encoding is transformed into motif embedding through convolutional layers. If sequence length is longer than the predefined threshold, an alternating combination of convolutional and pooling layers is used to obtain motif embedding. Then, the embeddings with absolute positional information are subsequently passed into the transformers for further processing. The output features from the transformers are used to conduct cell

classification. TransformerST[31] constructs a variational-transformers framework for data representation and employs CNN as both the decoder and encoder. It introduces a graph transformer between the decoder and encoder to analyze spatial transcriptomics data. By constructing an undirected graph, the graph transformer is able to learn nonlinear mappings and aggregate neighbor relationships. It makes high-resolution reconstruction of gene expression possible.

3.1.3 Single-cell language model based on single-cell multi-omics

The SCMVP[32] is a deep generative model based on transformers specifically designed for the simultaneous analysis of scRNA-seq and scATAC-seq data. The model establishes two independent channels at the encoder and decoder layers for processing scRNA data and scATAC data. In the scRNA channel, the masked attention mechanism is adopted, while in the scATAC channel, the self-attention mechanism is employed. Subsequently, the outputs of the two channels are combined, and the mean and variance of the common latent variables are obtained through a shared linear layer. scMoFormer[33] is a multimodal model based on transformers that uses a heterogeneous graph to model single-cell data. It constructs a multimodal heterogeneous graph containing three types of nodes: cells, genes, and proteins. In the training framework, three transformers are used, each dedicated to extracting the data representation of the corresponding modality. Finally, a multi-layer fully connected network is utilized to predict the target protein expression level of each cell. DeepMAPS[34] is a model that introduces the Heterogeneous Graph Transformer (HGT) framework. It constructs a heterogeneous graph using a cell-gene matrix. Then, the entire heterogeneous graph is divided into multiple subgraphs and HGT is applied on these subgraphs. Subgraph sampling is performed through a sparse-based feature selection method. During training process, the information of nodes is updated through multiple iterations of training and the training on different subgraphs shares the same set of parameters. After training on all subgraphs is completed, HGT is applied to the entire heterogeneous graph to obtain data features. MarsGT[35] is an extended model based on DeepMAPS. The heterogeneous graph of MarsGT is constructed base on cell-gene matrix, gene-peak matrix, and cell-peak matrix. Compared to DeepMAPS, it is better to obtain features of single-cell data from the perspective of regulatory networks by increasing the peak. During the subgraph sampling stage, a probability-based subgraph sampling method is employed to select genes and regulatory regions associated with rare cells. Then, the model is trained on the subgraph using transformers. After obtaining the trained weights, the pre-trained model is applied to the entire graph for training.

3.2 Single-cell large language model

Currently, large language models are also being applied to single-cell domains. The GPT and BERT have emerged as leading representatives. This section provides an introduction of the current single-cell large language models.

3.2.1 Single-cell large language model based on single-cell transcriptomics

The scBERT[36] is the first single-cell pre-training model constructed based on the BERT architecture. The structure of scBERT is shown in Fig. 2. During the training process, scBERT has been optimized to eliminate of artificial biases and overfitting for enhancing the generalization capability of model. To capture the similarity between genes, the scBERT employs the gene2vec[37] to obtain gene embedding for each gene. The input embedding information is obtained to capture relationships between genes by combining expression embedding and gene embedding. The embedding design allows scBERT to more effectively transform gene expression information into the input for the transformers to generate cell-specific embedding. Considering that most scRNA-seq data dimensions exceed the 512-limitation of transformers, scBERT utilizes the performer to reduce computational complexity through approximate self-attention calculations, which employs a linear attention mechanism based on low-rank random feature mapping. It enables scBERT to input over 16 000 genes when processing long sequence data. In addition, the scBERT also provides the interpretability by using Enrichr to visual attention weight to reflect the contribution of genes.

The scFoundation[38] is a large pre-trained model based on transformers with 100 million parameter scale. The embedding module of scFoundation is employed to get final embeddings with positional information. In addition, scFoundation adopts an

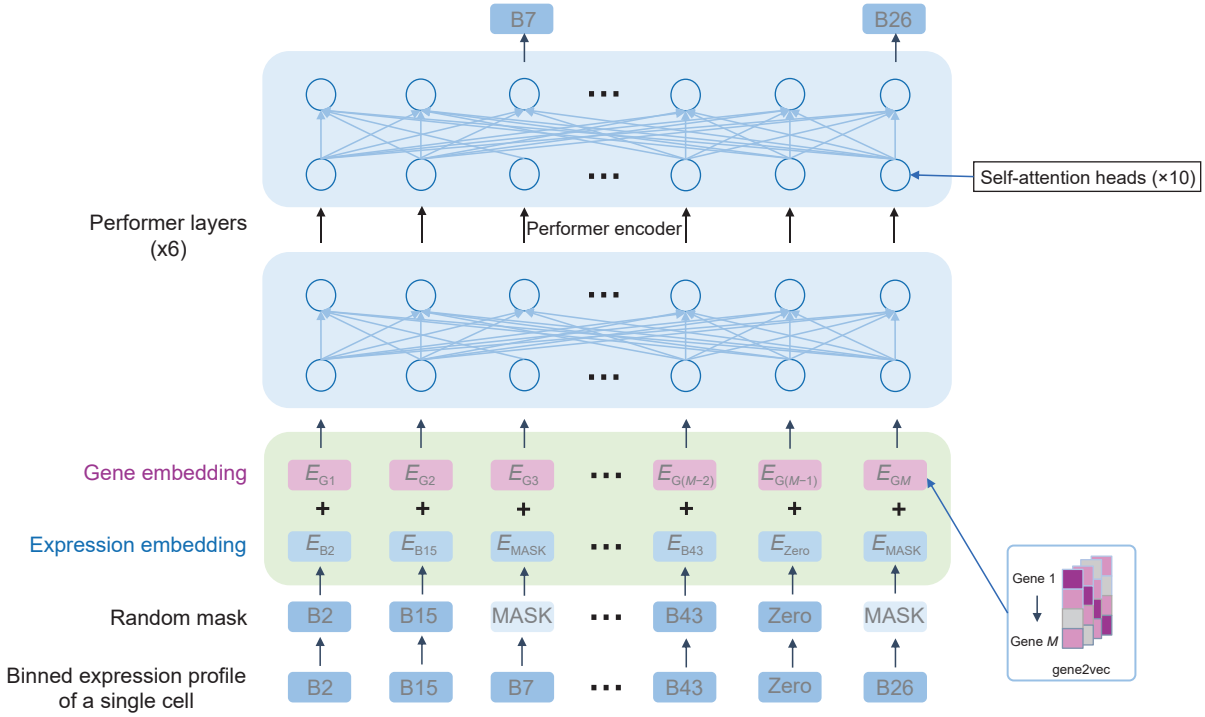


Fig. 2 Framework of scBERT. B* represents a discretized expression converted from scRNA-seq data and is randomly masked; E_{B*} represents the expression embedding of each gene; and E_{G*} is obtained by gene2vec and represents the gene embedding of each gene.

asymmetric encoder-decoder architecture. During the encode phase, it exclusively conducts the training on non-zero and non-masked expressed genes to reduce computational costs. In the decode phase, it restores zero and masked expressed genes to learn relationships among all genes. The read-depth-aware task is utilized as training strategy to train a pre-trained model, which is illustrated in Fig. 3. It successfully harmonizes read-depth differences across different cells to prove more coordinated and precise when dealing with cells with varying sequencing depths.

3.2.2 Single-cell large language model based on single-cell multi-omics

The scGPT[39] is the first single-cell foundation model based on transformers that undergone generative pre-training on over 33 million cells. The model draws inspiration from GPT. The structure of scGPT is depicted in Fig. 4. scGPT treats genes as tokens and uses a condition token to represent the positional information of genes. In addition, it employs value binning to address differences between different sequencing batches. scGPT uses stacked transformers, layers and flash-attention[40] to handle single-cell

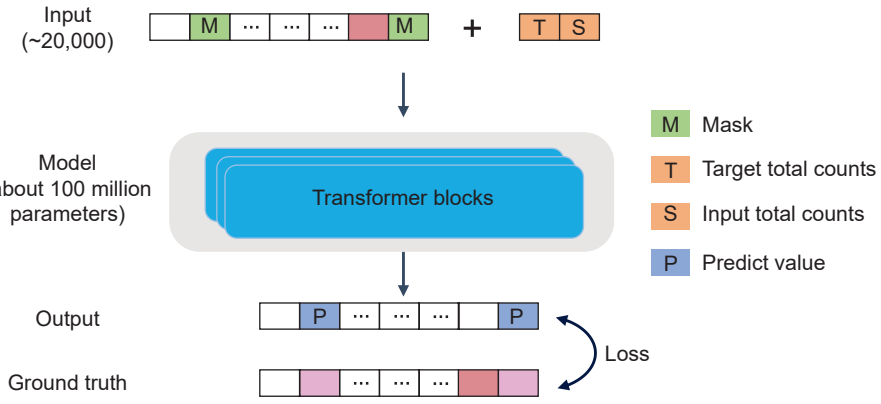


Fig. 3 Pre-training module structure of scFoundation.

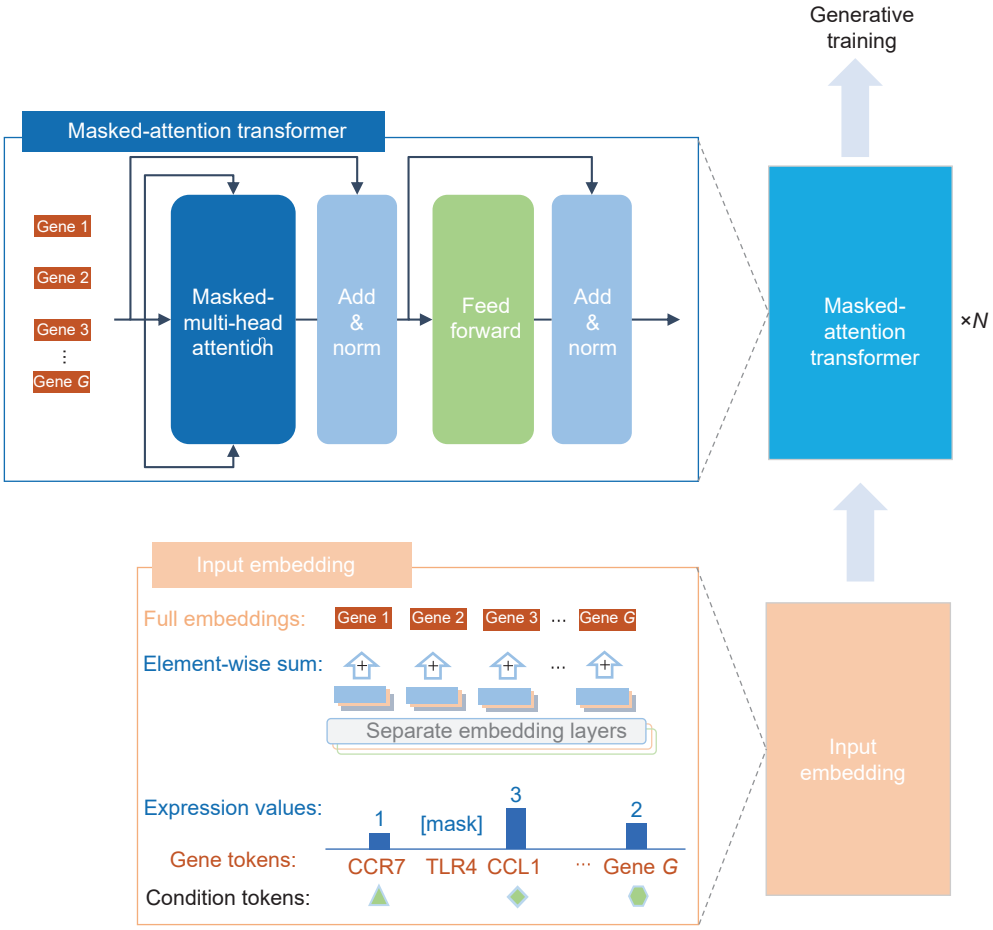


Fig. 4 Framework of scGPT.

multi-omics data. Flash-attention can effectively address the sequence length limitation and reduce computational cost. In terms of interpretability, scGPT focuses on key genes through pre-training on a good deal of single-cell data. Therefore, it has more comprehensive interpretability. While scGPT demonstrates impressive performance, It still has some shortcomings. It proves competitive in low-data settings, but it requires careful consideration of experimental conditions in zero-shot settings. Moreover, the current pre-training methods may lack universal applicability.

The CellPLM[41] is the first single-cell pre-trained model based on transformers that considers the relationship between cells. The structure of CellPLM is depicted in Fig. 5. It establishes a gene expression embedder for processing input data. The embedder initializes an embedding vector for each type of gene and filters out unmeasured genes and randomly masked genes. The gene expression embedder aggregates gene embedding based on their expression levels in each cell, and then transforms them into a suitable input of the transformers. These expression embeddings are then input into a structure of an encoder-decoder by utilizing a latent space between the encoder and decoder. The encoder part comprises N transformers blocks. However, the computational complexity of transformers exhibits quadratic growth which results in significant computational costs[42]. CellPLM replaces the transformers with a variant called Flowformer[43] to resolve the input constraints and computational complexity problems associated with the transformers. To more effectively capture cell-cell relationships and spatial positional information of individual cells, CellPLM incorporates Spatial Resolution Transcriptome (SRT) data into the encoder for training. SRT data contain position embedding information. The position embedding are combined with expression embedding to obtain the final input embedding. In the latent space, a Gaussian mixture model is employed. The decoder employs several feedforward layers (FFLayers) to train latent space vectors and acquires

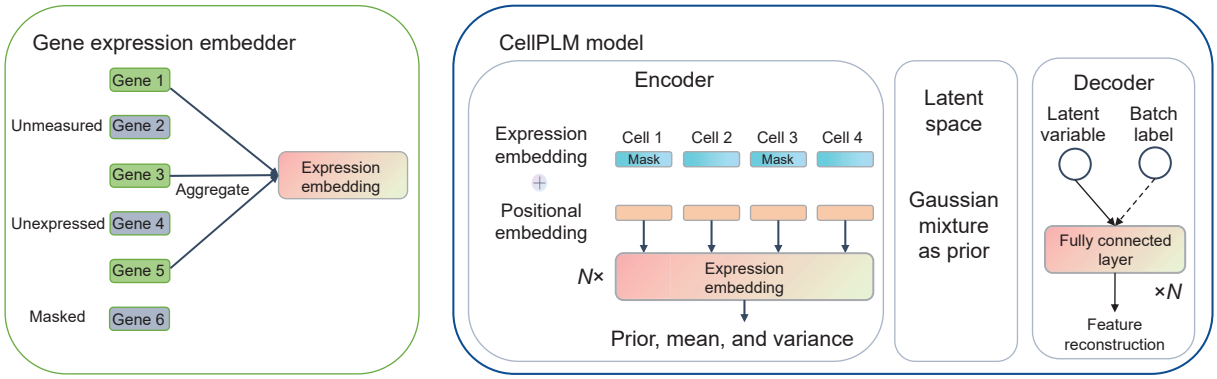


Fig. 5 Training framework of CellPLM.

the batch label of each cell by learning from the learnable lookup table.

3.2.3 Single-cell large language model based on gene expression ranking

The tGPT[44] is an autoregressive unsupervised training model based on transformers. It utilizes the ranking of gene expression to predict the index of the next gene. Gene expression ranking provides the relative position of genes and is more suitable for large-scale gene screening and comparative analysis. However, this strategy may only consider genes with higher expression levels and neglect the specific information contained in low-expression genes. The structure of tGPT is depicted in Fig. 6. The tGPT predefines a length limit of input sequence and any part of the input sequence exceeding this limit is truncated, while the sections not reaching the limit are padded as 0. In the training process, it combines gene token embedding with positional encoding embedding. Final embedding undergoes 8 transformers, modules to extract features from single-cell sequences.

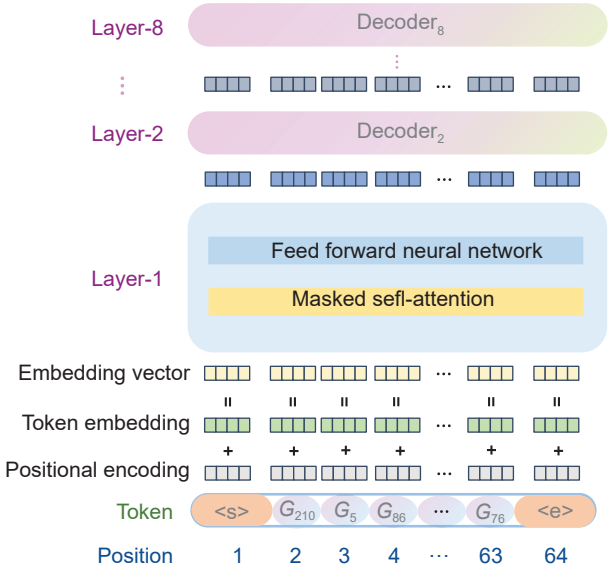


Fig. 6 Module structure of tGPT.

The Cell2Sentence (C2S)[45] is a pretrained model fine-tuned on GPT-2, focusing on handling text sequences containing gene names. Through fine-tuning, C2S is capable of generating new cell sentences and reversely converting them back into gene expression vectors, retaining most of the information. The order of gene names is determined by the expression ranking of each gene and C2S uses these gene name sequences as its input. By converting cell text sequences back into gene expressions, C2S minimizes information loss and retains key information from the original data in most cases. This method enables transformers to acquire information about single-cell data, but the sequence conversion operation often results in higher computational costs.

4 Downstream Task Analysis

The single-cell language models based on transformers have conducted on various downstream tasks, including batch correction, cell clustering, cell type annotation, gene network inference, and perturbation responses. The datasets used for these downstream tasks are primarily obtained through databases, such as TCGA[46] and GEO[47]. The details of them are shown in Table 1.

4.1 Batch correction

With the increasing quantity of single-cell data, the variability between different batches has become an increasingly significant interference in data analysis. It becomes an urgent challenge to improve the effectiveness of batch correction. Three key metrics are used to evaluation of batch correction effects including

Table 1 Details of downstream multi-task (single-cell large language models are marked with an asterisk).

Downstream task	Model	Metric	Dataset
Batch correction	tGPT*	kBET	HCA[48]
	scGPT*	ASWbatch, GraphConn	COVID-19[49], PBMC 10[50], Perirhinal Cortex[51]
Cell clustering	scMVP	ARI	Paired-seq cell line data[52], SNARE-seq cell line data[53]
	tGPT*	ARI, NMI, FMI	HCA[48], HCL[54], TCGA[55], Macaque Retina[56], GTEx[57], Tabula Muris[58]
	CellPLM*	ARI, NMI	public dataset[59]
	DeepMAPS	ASW, ARI	PBMC[50], lung tumor leukocytes CITE-seq dataset[60]
Cell type annotation	TransCluster	F1-score, Precision, Recall, MCC	Shao[61], Baron[62]
	PROTRAIT	ARI, AMI	sci-ATAC human atlas[63]
	scBERT*	Accuracy, ARI, F1-score	Baron[62], Muraro[64], Segerstolpe[65], Xin[66]
	scGPT*	Accuracy, Precision, Recall, F1-score	hPancreas[67], multiple sclerosis[68], tumor-infiltrating myeloid[69]
	CellPLM*	Precision, F1-score	hPancreas[67], multiple sclerosis[68]
Gene network inference	DeepMAPS	Closeness centrality, Eigenvector centrality, Functional enrichment analysis	Reactome[70], DoRothEA[71], TRRUST v2[72]
	scGPT*	Pathway enrichment analysis	Immune Human[73], ChIP-994Atlas database[74], Adamson[75]
Perturbation prediction	scFoundation*	MSE	Dixit[76], Adamson[75], Norman[77]
	scGPT*	PCC	Adamson[75], Norman[77]
	CellPLM*	RMSE	Adamson[75], Norman[77]

k-nearest neighbor Batch Effect Test (kBET)[78], Average Silhouette Width for batch correction (ASWbatch)[79], and Graph Connectivity measurement (GraphConn)[80]. The kBET assesses the effectiveness of correction by comparing the distribution of cells within and between batches. Its acceptance rate reflects the uniformity of cell distribution after correction. A higher acceptance rate indicates the preservation of biological heterogeneity and a reduction in technical batch effects. The ASWbatch originates from the concept of silhouette width in cluster analysis. It is used to measure the clustering effect after removing batch effects. The GraphConn is a method for evaluating the connectivity between cells in the dataset after batch correction. It aims to quantify the enhancement of cell-to-cell connectivity post-correction for reflecting the reduction of batch effects.

The tGPT[44] adopts the ranking of gene expression to void the interference of actual expressions of Highly Variable Genes (HVGs) and batch information during training. It is trained on the HCA dataset[48], utilizing the kBET acceptance rate to reflect the magnitude of differences between different batches. In addition, tGPT conducts an Immune Checkpoint Blockade (ICB) clinical trial. By quantifying the expression features of different attention heads, it is demonstrated that these attention heads have prognostic significance in this clinical trial. scGPT[39] conducts batch effect experiments by fine-tuning on pre-trained models. To quantify batch correction performance, scGPT calculates the Average Silhouette Width (ASW_{batch}) and GraphConn[81]. It computes the AvgBATCH (i.e, average of ASWbatch and GraphConn), to comprehensively represent batch performance. scGPT evaluates batch correction performance on three datasets, including COVID-19[49], PBMC 10[50], and Perirhinal Cortex[51]. The evaluation is conducted against three methods including Seurat[82], Harmony[83], and scVI[84]. scGPT achieves a best performance AvgBATCH value on the three datasets. However, scGPT does not achieve excellent batch effect correction in zero-shot settings[85].

4.2 Cell clustering

The goal of cell clustering analysis is to group cells based on their gene expression patterns. When evaluating the accuracy of clustering results, commonly used metrics include Adjusted Rand Index (ARI)[86], Average Silhouette Width (ASW)[79], and

Normalized Mutual Information (NMI)[87]. ARI adjusts the rand index by comparing the observed pair-wise concordance to the expected random concordance, and yields a measure of clustering consistency. ASW measures the difference in similarity between samples and different clusters by calculating the silhouette width for each sample. It offers a intuitive evaluation of clustering results. NMI utilizes normalized mutual information to eliminate the influence of the number of clusters and the total number of samples, which makes it useful for comparing clustering results under different parameter settings.

The scMVP[32] employs a joint deep learning model to learn features from both scATAC data and scRNA data. It is trained on Paired-seq cell line data[52] and SNARE-seq cell line data[53]. Then it utilizes Uniform Manifold Approximation and Projection (UMAP) visualization to perform cell clustering analysis on cell clusters. It successfully identifies different numbers of cell subpopulations and effectively separates the integration data of scRNA-seq and scATAC-seq. It confirms its effectiveness in cell clustering analysis. tGPT[44] is applicable to large-scale tissue samples through pre-training. It partitions samples into distinct clusters that correspond to different organs. It is trained on six datasets, including HCA[48], HCL[54], TCGA[55], Macaque Retina[56], GTEx[57], and Tabula Muris[58]. The experimental results demonstrate that it achieves excellent performance in cell clustering tasks. CellPLM[41] conducts unsupervised clustering analysis by extracting cell embedding vectors from the dataset without fine-tuning. CellPLM achieves zero-shot clustering experiments on a public dataset[59]. It compares with PCA, Geneformer, and scGPT. In the experiments, it achieves the highest ARI and NMI. DeepMAPS[34] validates cell clustering on ten single-cell multi-omics datasets. It trains with 36 parameter combinations and compares with Seurat, MOFA+[88], TotalVI[89], and Harmony. In all experiments, DeepMAPS achieves the best ARI and ASW. Furthermore, DeepMAPS performs single-cell multi-omics integration analysis on the PBMC dataset[50] and the CITE-seq dataset of lung tumor leukocytes[60]. It successfully identifies 13 cell types and validates its effectiveness.

4.3 Cell type annotation

Cell type annotation refers to assigning known cell type labels to each cell or cell cluster, which aids in gaining a deeper understanding of the biological significance of the cells[90]. When evaluating the performance of cell annotation, commonly used metrics include precision, recall, accuracy, and F1-score[91]. Precision represents the proportion of correctly predicted samples of a specific category among all samples predicted as that category by the model. Accuracy denotes the ratio of correctly classified samples to the total number of samples. Recall indicates the proportion of true samples of a specific category that the model correctly identifies as that category. The F1-score is the harmonic mean of precision and recall. It offers a comprehensive evaluation of model performance.

The TransCluster[24] is the first model to apply transformers to cell type annotation. It is trained on the Shao dataset[61] and the Baron dataset[62], and demonstrates efficient performance in cell type prediction tasks. PROTRAIT[30] is trained on the sci-ATAC human atlas[63] and generates cell embeddings that reflect the distribution of the scATAC-seq data. Then, it uses the *k*-Nearest Neighbors (KNN) for cell type annotation. scBERT[36] is pre-trained on 9 scRNA-seq datasets, then fine-tuning is performed on the trained model. Final, it uses the K-means algorithm to annotate cell types. scBERT performs cell annotation tasks on the Baron dataset[62], the Muraro dataset[64], the Segerstolpe dataset[65], and the Xin dataset[66]. Both scGPT[39] and CellPLM[41] are trained on the hPancreas[67] dataset and Multiple Sclerosis (MS)[68] dataset to perform cell annotation task. scGPT performs normalization, log transformation and binning operations on gene expression values, then cell type annotation is achieved through fine-tuning. In addition, scGPT is trained on the tumor-infiltrating myeloid dataset (Mye.)[69] and evaluated on query partitions of three previously unseen cancer types. The results indicate that scGPT has high accuracy in distinguishing immune cell subtypes. CellPLM adds a feedforward layer during the fine-tuning process and utilizes standard cross-entropy loss function for the fine-tuning process. Fine-tuned CellPLM exhibits a significant improvement in F1-score and precision metrics on the hPancreas dataset and Multiple Sclerosis (MS) dataset compared to the from-scratch CellPLM.

4.4 Gene network inference

Gene network inference analysis reveals regulatory associations between genes by comparing gene

expression patterns under different conditions. Currently, single-cell language models based on transformers have introduced innovative perspectives to the study of gene regulatory networks[92]. Centrality score metrics, including Closeness Centrality (CC) and Eigenvector Centrality (EC)[93], are used to the experiment of single-cell language models. The CC assesses the average distance of a gene node relative to other gene nodes in the network. EC considers not only the number of connections of a gene node but also the importance of the other gene nodes that it is connected to. In addition, functional enrichment analysis[94] and pathway enrichment analysis[95] are employed in experimental analysis. Functional enrichment analysis aims to identify biological functions or processes that are significantly enriched in a set of genes. Pathway enrichment analysis is similar to functional enrichment analysis but focuses more on known biochemical pathways[96]. It aims to deeply understand how genes function through synergistic interactions within specific biological pathways.

The DeepMAPS[34] uses the Steiner Forest Problem (SFP) to identify genes contributing significantly to cell cluster features and constructs a gene correlation network. It defines sets of genes regulated by the same Transcription Factor (TF) as regulons and compares regulon activities between cell clusters. Then, it selects regulons with significantly higher activity scores as cell-type-specific regulons and constructs Gene Regulatory Networks (GRNs) based on cell cluster regulons. After constructing GRNs, DeepMAPS conducts functional enrichment analysis. Specifically, it employs hypergeometric tests to compare the intersection of GRN results with regulons in the database and evaluates whether the predicted regulons in the GRN are enriched for the same functions or pathways as known regulons. DeepMAPS is trained on single-cell multi-omics datasets from the 10X database. The experiment of DeepMAPS demonstrates that the GRNs exhibit a greater number of distinct TFs and cell-type-specific regulons, and they are enriched in specific functions or pathways. In addition, scGPT[39] demonstrates high interpretability in gene regulatory network experiments. Pre-training enables scGPT to emphasize genes with intricate relationships. It improves the interpretability of scGPT. In the Human Leukocyte Antigen (HLA) dataset, scGPT forms an HLA gene network through zero-shot learning. On the Immune Human dataset[73], fine-tuned scGPT generates CD gene networks through zero-shot learning and visualization of the gene information. scGPT performs pathway enrichment analysis on the Reactome database[70]. It successfully validates the extracted gene program and identifies 22 additional pathways. These experiments demonstrate the ability of scGPT to capture complex gene relationships. Through pre-training and fine-tuning, scGPT achieves stronger generalization capabilities.

4.5 Perturbation responses

Single-cell perturbation prediction experiments aim to predict and analyze the biological responses of cells to external stimuli or changes introduced into single cells[97]. Mean Squared Error (MSE) and Root Mean Squared Error (RMSE) have become two important metrics for evaluating the performance of model in predicting how cells respond to specific perturbations[3]. MSE is used to measure the accuracy of the model in predicting the response of single cells to specific perturbations. A lower MSE value indicates that the predictions of model are more consistent with the actual observed values. RMSE is the square root of MSE and provides an error measure in the same units as the original data. It directly reflects the magnitude of the prediction error.

In perturbation responses prediction experiments, scFoundation[38] is combined with the GEARS[98] to construct personalized gene co-expression graphs for each cell. It significantly improves the accuracy of gene perturbation predictions. It is evaluated on three datasets, including the Dixit dataset[76], the Adamson dataset[75], and the Norman dataset[77]. It obtains lower MSE values. In addition, scGPT[39] uses the pre-trained parameters of embedding and transformer layers to initialize fine-tuning. The fine-tuning process uses genes with zero and non-zero expression. scGPT is compared with GEARS and CPA[99] on the Adamson dataset and the Norman dataset. It accurately predicts the expression changes of the top 20 Differentially Expressed (DE) genes in the datasets. During the fine-tuning process, CellPLM[41] initializes other components except the decoder with pre-trained weights. CellPLM is compared with GEARS and scGen[100] on the Adamson dataset and the Norman dataset. It conducts two types of experiments (single-gene perturbation and double-gene perturbation) on the Norman dataset and only single-gene perturbation on the Adamson dataset. In each experiment, CellPLM

exhibites lower RMSE than GEARS and scGen.

5 Challenge and Prospect

In the field of single-cell research, transformers contribute to a deeper understanding of these vast and complex datasets. It enhances the simulation and comprehension of cellular processes. In this section, we discuss the challenges encountered by transformers-based single-cell models. We focus primarily on limitations in the transformer-based single-cell language model, including handling long sequence data, overfitting risks in pre-training, and computational requirement and interpretability. In addition, we also analyze some potential future research directions.

5.1 Sequence data processing

The transformers-based single-cell language models have strong representational capabilities on single-cell sequence data. However, single-cell sequence data often contains excessively long sequences[101]. It leads to an exponential increase in the computational complexity of these models. In addition, single-cell data with long sequences may contain more complex gene relationships. Nevertheless, the self-attention mechanism of the transformers tends to capture dependencies between adjacent positions in the sequence. It may causes the model to ignore some key gene information. scBERT[36] adopts a variant of transformers, called "Performer", to solve the problem. scBERT uses the low-rank attention mechanism of Performer to avoid over-focusing on dependencies between adjacent positions. When dealing with sparse DNA sequences, the attention mechanism of Performer may exhibit better robustness. Although Performer achieves good results, there are certain challenges in terms of data precision and sensitivity to model parameters due to the low-rank attention mechanism. In addition, the effectiveness of Performer is not always superior to that of the traditional transformers for different datasets and tasks. However, it is undeniable that using some variants of the transformers has brought new insights to the research of single-cell language models.

5.2 Overfitting risks in pre-training

Although transformers-based single-cell language models are increasingly inclined to adopt pre-training techniques, the analysis of these pre-trained models in terms of overfitting issues is relatively limited. The characteristic of single-cell data lies in its diversity of types and different types of single-cell data may vary significantly. It may lead to an imbalanced distribution of pre-train samples, and potentially causing overfitting on smaller datasets. To address this issue, data augmentation techniques can be introduced into the pre-training. Currently, Generative Adversarial Networks (GAN) have shown promising results in the field of single-cell data augmentation[102]. By using GAN to generate synthetic data samples that are similar to the original data, the diversity of the dataset can be effectively increased. It can mitigate the overfitting problems caused by data imbalance. In addition, interpolating and extrapolating between original single-cell data samples can also be considered. By using methods, such as linear interpolation, polynomial interpolation, or deep learning models, to generate new samples, the quantity and diversity of the data can be increased. It further enhances the generalization capability and robustness of models. We believe that incorporating these methods into the pre-training process of single-cell language models may help address the issue of overfitting in the models.

5.3 Computing requirement

Currently, transformers-based single-cell language models for single-cell multi-omics research are still in their early stages. Future work may involve incorporating more omics data in the pre-training phase to study single-cell multimodal tasks. However, the incorporation of omics data has led to an even larger scale of data. It causes challenges related to computational costs. Recently, the combination of recurrent neural networks and transformers has reduced computational costs by speeding up the training of transformers[103]. This method could be considered as a possibility for application in single-cell language models. In addition, the parallel computing capabilities of transformers still face challenges. In the self-attention mechanism, the attention weights for each position need to be calculated sequentially and cannot be directly parallelized. When processing batch data, the sequence lengths of different single-cell samples may vary, increasing the complexity of parallel computing. In the future, solving the parallel computing capabilities of single-cell language models may become increasingly critical.

5.4 Interpretability

Transformers-based single-cell language models offer significant advantages in terms of interpretability. They are capable of assigning different gene weights during the processing of sequence data to identify key features in the representation process. In single-cell research, the capability is crucial for understanding complex biological processes, such as gene expression, protein interactions, and gene regulation[104]. In addition, single-cell data are highly complex and diverse. Each cell potentially exhibits unique gene expression pattern[105]. Through the self-attention mechanism, transformers have successfully provided interpretability for the predictions of the key features. This helps biologists understand how models assign weights to different genes or cells, and gain insights into gene expression patterns. Although transformers-based single-cell language models have achieved good results, these models still employ a black-box training approach. It inevitably affects the application of models in clinical settings. Therefore, improving the interpretability of single-cell language models remains a challenging research problem.

5.5 Validation analysis

The single-cell language models and single-cell large language models mentioned in this paper have demonstrated promising results in experiments. Currently, some of these models have been subjected to benchmark experiments[106–108], which have revealed that different models exhibit varying performance across different tasks. These models have been proven to have the capability to integrate representations from diverse single-cell omics data. In particular, pre-trained models like scGPT have shown remarkable performance in gene function prediction tasks and achieve good results even without fine-tuning. However, the application of single-cell language models and single-cell large language models is still in its early stages and their generalizability faces certain challenges. In addition, comparing with some of the latest methods, such as Sccross[109] and ctpredictor[110], will also help to promote research progress. Therefore, we provide an accessible link to the experimental code of the single-cell language model, please refer to Table 2 for details. We hope these resources can provide some assistance to researchers who are interested in this field.

6 Conclusion

The transformer-based single-cell language model has shown promising results in single-cell data analysis. In this review, we provide a detailed overview of single-cell language models and single-cell large language models. We summarize the methods of these models as well as their applications in downstream tasks. While these models may not achieve optimal performance in certain evaluation metrics, they hold potential contributions and applications in single-cell research. They open new possibilities for research and

Table 2 Link to the code of the models.

Model	Input data type	Data repositories address
TransCluster	scRNA-seq	https://github.com/Danica123/TransCluster.git
scTransSort	scRNA-seq	https://github.com/jiaojiao-123/scTransSort
CIForm	scRNA-seq	https://github.com/zhanglab-wbgcas/CIForm
STGRNS	scRNA-seq	https://github.com/zhanglab-wbgcas/STGRNS
T-GEM	scRNA-seq, transcriptomics (the pan-cancer RNA-Seq)	https://github.com/TingheZhang/TGEM
PROTRAIT	scATAC-seq	https://github.com/ZhangLab312/PROTRAIT
scMVP	scRNA-seq, scATAC-seq	https://github.com/bm2-lab/scMVP
scMoFormer	scRNA-seq, Proteomics	https://github.com/OmicsML/scMoFormer
DeepMAPS	scRNA-seq, scATAC-seq, CITE-seq	https://github.com/OSU-BMBL/deepmaps
MarsGT	scRNA-seq, scATAC-seq	https://github.com/mtduan/marsgt
scBERT	scRNA-seq	https://github.com/TencentAILabHealthcare/scBERT
scFoundation	scRNA-seq	https://github.com/biomapresearch/scFoundation
scGPT	scRNA-seq	https://github.com/bowang-lab/scGPT
CellPLM	scRNA-seq, spatial transcriptomics, Perturb-seq	https://github.com/OmicsML/CellPLM
tGPT	scRNA-seq	https://github.com/deeplearningplus/tGPT
Cell2Sentenc	scRNA-seq	https://github.com/vandijklab/cell2sentence-ft

applications in the field and present significant avenues for further development. We think that the potential areas for improvement may include refining data preprocessing methods, reducing computational costs, enhancing model interpretability, and optimizing the transfer learning process. In-depth investigations into these directions will facilitate more effective utilization of various types of single-cell data. This review aims to provide an overview of single-cell language models and hope promoting progress in the field of single-cell research.

Acknowledgment

This work was supported by the National Natural Science Foundation of China (No. 62072124), the Natural Science Foundation of Guangxi (No. 2023JJG170006), the Natural Science and Technology Innovation Development Foundation of Guangxi University (No. 2022BZRC009), the CAAI-Huawei MindSpore Open Fund (No. CAAIXSJLJJ-2022-022A), the Project of Guangxi Key Laboratory of Eye Health (No. GXYJK-202407), and the Project of Guangxi Health Commission Eye and Related Diseases Artificial Intelligence Screen Technology Key Laboratory (No. GXYAI-202402).

References

[1] W. Lan, T. Yang, Q. Chen, S. Zhang, Y. Dong, H. Zhou, and Y. Pan, Multiview subspace clustering via low-rank symmetric affinity graph, *IEEE Trans. Neural Networks Learn. Syst*., doi: 10.1109/TNNLS.2023.3260258.

[2] K. Makhani, X. Yang, F. Dierick, N. Subramaniam, N. Gagnon, T. Ebrahimian, H. Wu, J. Ding, and K. K. Mann, Unveiling the impact of arsenic toxicity on immune cells in atherosclerotic plaques: Insights from single-cell multi-omics profiling, bioRxiv, doi: 10.1101/2023.11.23.568429.

[3] W. Lan, J. Chen, M. Liu, Q. Chen, J. Liu, J. Wang, and Y. P. Chen, Deep imputation bi-stochastic graph regularized matrix factorization for clustering single-cell RNA-sequencing data, *IEEE/ACM Trans. Comput. Biol. Bioinf*., doi: 10.1109/TCBB.2024.3387911.

[4] T. Stuart and R. Satija, Integrative single-cell analysis, *Nat. Rev. Genet.*, vol. 20, no. 5, pp. 257–272, 2019.

[5] W. B. Cavnar and J. M. Trenkle, N-grambased text categorization. in *Proc. SDAIR-94, 3rd Annu. Symp. Document Analysis and Information Retrieval*, Las Vegas, NV, USA, 1994, p. 14.

[6] S. R. Eddy, Hidden Markov models, *Curr. Opin. Struct. Biol.*, vol. 6, no. 3, pp. 361–365, 1996.

[7] R. Petegrosso, Z. Li, and R. Kuang, Machine learning and statistical methods for clustering single-cell RNA-sequencing data, *Briefings Bioinf.*, vol. 21, no. 4, pp. 1209–1223, 2020.

[8] W. Zaremba, I. Sutskever, and O. Vinyals, Recurrent neural network regularization, arXiv preprint arXiv: 1409.2329, 2014.

[9] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, Gradient-based learning applied to document recognition, *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[10] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, Attention is all you need, in *Proc. 31st Int. Conf. Neural Information Processing Systems*, Long Beach, CA, USA, 2017, pp. 6000–6010.

[11] Z. Dai, B. Cai, Y. Lin, and J. Chen, UP-DETR: Unsupervised pre-training for object detection with transformers, in Proc. *IEEE/CVF Conf. Computer Vision and Pattern Recognition*, Nashville, TN, USA, 2021, pp. 1601–1610.

[12] Z. Wang, Z. Deng, F. Liu, Y. Huang, H. Yu, and J. Cui, OSNet and MNetO: Two types of general reconstruction architectures to transform DBP images for linear computed tomography in multi-scenarios, *IEEE Trans. Instrum. Meas*., vol. 73, p. 4505016, 2024.

[13] A. T. Liu, S. W. Li, and H. Y. Lee, TERA: Self-supervised learning of transformer encoder representation for speech, *IEEE/ACM Trans. Audio Speech Lang. Process*., vol. 29, pp. 2351–2366, 2021.

[14] Z. Dai, Z. Yang, Y. Yang, J. Carbonell, Q. Le, and R. Salakhutdinov, Transformer-XL: Attentive language models beyond a fixed-length context, in *Proc. 57th Annu. Meeting of the Association for Computational Linguistics*, Florence, Italy, 2019, pp. 2978–2988.

[15] N. Kitaev, Ł. Kaiser, and A. Levskaya, Reformer: The efficient transformer, in *Proc. 8th Int. Conf. Learning Representations*, Addis Ababa, Ethiopia, 2020. arXiv:2001.04451

[16] T. Young, D. Hazarika, S. Poria, and E. Cambria, Recent trends in deep learning based natural language processing, *IEEE Comput. Intell. Mag.*, vol. 13, no. 3, pp. 55–75, 2018.

[17] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, Improving language understanding by generative pre-training, https://openai.com/research/language-unsupervised, 2023.

[18] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in *Proc. 2019 Conf. North American Chapter of the Association for Computational Linguistics*: *Human Language Technologies*, Minneapolis, MN, USA, 2019, pp. 4171–4186.

[19] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov, and Q. V. Le, XLNet: Generalized autoregressive pretraining for language understanding, in *Proc. 33rd Int. Conf. Neural Information Processing Systems*, Vancouver, Canada, 2019, p. 517.

[20] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, RoBERTa: A robustly optimized BERT pretraining approach, arXiv preprint arXiv: 1907.11692, 2019.

[21] R. Xiong, Y. Yang, D. He, K. Zheng, S. Zheng, C. Xing, H. Zhang, Y. Lan, L. Wang, and T. Y. Liu, On layer

normalization in the transformer architecture, in *Proc. 37th Int. Conf. Machine Learning*, Virtual Event, 2020, pp. 10524–10533.

[22] K. Wu, H. Peng, M. Chen, J. Fu, and H. Chao, Rethinking and improving relative position encoding for vision transformer, in *Proc. IEEE/CVF Int. Conf. Computer Vision*, Montreal, Canada, 2021, pp. 10013–10021.

[23] O. I. Abiodun, A. Jantan, A. E. Omolara, K. V. Dada, A. M. Umar, O. U. Linus, H. Arshad, A. A. Kazaure, U. Gana, and M. U. Kiru, Comprehensive review of artificial neural network applications to pattern recognition, *IEEE Access*, vol. 7, pp. 158820–158846, 2019.

[24] T. Song, H. Dai, S. Wang, G. Wang, X. Zhang, Y. Zhang, and L. Jiao, TransCluster: A cell-type identification method for single-cell RNA-SEQ data using deep learning based on transformer, *Front. Genet.*, vol. 13, p. 1038919, 2022.

[25] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*: *Data Mining, Inference, and Prediction*, 2nd ed. New York, NY, USA: Springer, 2009.

[26] L. Jiao, G. Wang, H. Dai, X. Li, S. Wang, and T. Song, scTransSort: Transformers for intelligent annotation of cell types by gene embeddings, *Biomolecules*, vol. 13, no. 4, p. 611, 2023.

[27] J. Xu, A. Zhang, F. Liu, L. Chen, and X. Zhang, CIForm as a transformer-based model for cell-type annotation of large-scale single-cell RNA-seq data, *Briefings Bioinf.*, vol. 24, no. 4, p. bbad195, 2023.

[28] J. Xu, A. Zhang, F. Liu, and X. Zhang, STGRNS: An interpretable transformer-based method for inferring gene regulatory networks from single-cell transcriptomic data, *Bioinformatics*, vol. 39, no. 4, p. btad165, 2023.

[29] T. H. Zhang, M. M. Hasib, Y. C. Chiu, Z. F. Han, Y. F. Jin, M. Flores, Y. Chen, and Y. Huang, Transformer for gene expression modeling (T-GEM): An interpretable deep learning model for gene expression-based phenotype predictions, *Cancers*, vol. 14, no. 19, p. 4763, 2022.

[30] Z. Wang, Y. Zhang, Y. Yu, J. Zhang, Y. Liu, and Q. Zou, A unified deep learning framework for single-cell ATAC-seq analysis based on ProdDep transformer encoder, *Int. J. Mol. Sci.*, vol. 24, no. 5, p. 4784, 2023.

[31] C. Zhao, Z. Xu, X. Wang, K. Chen, H. Huang, and W. Chen, Transformer enables reference free and unsupervised analysis of spatial transcriptomics, bioRxiv, doi: 10.1101/2022.08.11.503261.

[32] G. Li, S. Fu, S. Wang, C. Zhu, B. Duan, C. Tang, X. Chen, G. Chuai, P. Wang, and Q. Liu, A deep generative model for multi-view profiling of single-cell RNA-seq and ATAC-seq data, *Genome Biol.*, vol. 23, no. 1, p. 20, 2022.

[33] W. Tang, H. Wen, R. Liu, J. Ding, W. Jin, Y. Xie, H. Liu, and J. Tang, Single-cell multimodal prediction via transformers, in *Proc. 32nd ACM Int. Conf. Information and Knowledge Management*, Birmingham, UK, 2023, pp. 2422–2431.

[34] A. Ma, X. Wang, C. Wang, J. Li, T. Xiao, J. Wang, Y. Li, Y. Liu, Y. Chang, D. Wang, et al., DeepMAPS: Single-cell biological network inference using heterogeneous graph transformer, BioRxiv, doi: 10.1101/2021.10.31.466658.

[35] X. Wang, M. Duan, J. Li, A. Ma, G. Xin, D. Xu, Z. Li, B. Liu, and Q. Ma, MarsGT: Multi-omics analysis for rare population inference using single-cell graph transformer, *Nat. Commun.*, vol. 15, no. 1, p. 338, 2024.

[36] F. Yang, W. Wang, F. Wang, Y. Fang, D. Tang, J. Huang, H. Lu, and J. Yao, scBERT as a large-scale pretrained deep language model for cell type annotation of single-cell RNA-seq data, *Nat. Mach. Intell.*, vol. 4, no. 10, pp. 852–866, 2022.

[37] Q. Zou, P. Xing, L. Wei, and B. Liu, Gene2vec: Gene subsequence embedding for prediction of mammalian N^6-methyladenosine sites from mRNA, *RNA*, vol. 25, no. 2, pp. 205–218, 2019.

[38] M. Hao, J. Gong, X. Zeng, C. Liu, Y. Guo, X. Cheng, T. Wang, J. Ma, X. Zhang, and L. Song, Large-scale foundation model on single-cell transcriptomics, *Nat. Methods*, doi: 10.1038/s41592-024-02305-7.

[39] H. Cui, C. Wang, H. Maan, K. Pang, F. Luo, N. Duan, and B. Wang, scGPT: Toward building a foundation model for single-cell multi-omics using generative AI, *Nat. Methods*, doi: 10.1038/s41592-024-02201-0.

[40] T. Dao, D. Y. Fu, S. Ermon, A. Rudra, and C. Ré, FLASHATTENTION: Fast and memory-efficient exact attention with IO-awareness, in *Proc. 36th Int. Conf. Neural Information Processing Systems*, New Orleans, LA, USA, 2022, p. 1189.

[41] H. Wen, W. Tang, X. Dai, J. Ding, W. Jin, Y. Xie, and J. Tang, CellPLM: Pre-training of cell language model beyond single cells, BioRxiv, doi: 10.1101/2023.10.03.560734.

[42] W. Lan, Y. Dong, H. Zhang, C. Li, Q. Chen, J. Liu, J. Wang, and Y. P. P. Chen, Benchmarking of computational methods for predicting circRNA-disease associations, *Briefings Bioinf.*, vol. 24, no. 1, pp. bbac613, 2023.

[43] Z. Huang, X. Shi, C. Zhang, Q. Wang, K. C. Cheung, H. Qin, J. Dai, and H. Li, FlowFormer: A transformer architecture for optical flow, in *Proc. 17th European Conf. Computer Vision*, Tel Aviv, Israe, 2022, pp. 668–685.

[44] H. Shen, J. Liu, J. Hu, X. Shen, C. Zhang, D. Wu, M. Feng, M. Yang, Y. Li, Y. Yang, et al., Generative pretraining from large-scale transcriptomes for single-cell deciphering, *iScience*, vol. 26, no. 5, p. 106536, 2023.

[45] D. Levine, S. A. Rizvi, S. Lévy, N. Pallikkavaliyaveetil, R. Wu, Z. Zheng, A. O. Fonseca, X. Chen, S. Ghadermarzi, R. M. Dhodapkar, et al., Cell2Sentence: Teaching large language models the language of biology, BioRxiv, doi: 10.1101/2023.09.11.557287.

[46] K. Tomczak, P. Czerwińska, and M. Wiznerowicz, Review the cancer genome atlas (TCGA): An immeasurable source of knowledge, *Contemp. Oncol.*, vol. 19, no. 1A, pp. A68–A77, 2015.

[47] T. Barrett, S. E. Wilhite, P. Ledoux, C. Evangelista, I. F.

Kim, M. Tomashevsky, K. A. Marshall, K. H. Phillippy, P. M. Sherman, M. Holko, et al., NCBI GEO: Archive for functional genomics data sets—update, *Nucleic Acids Res*., vol. 41, no. D1, pp. D991–D995, 2012.

[48] A. Regev, S. Teichmann, O. Rozenblatt-Rosen, M. Stubbington, K. Ardlie, I. Amit, P. Arlotta, G. Bader, C. Benoist, M. Biton, et al., The human cell atlas white paper, arXiv preprint arXiv: 1810.05192, 2018.

[49] M. Lotfollahi, M. Naghipourfar, M. D. Luecken, M. Khajavi, M. Büttner, M. Wagenstetter, Ž. Avsec, A. Gayoso, N. Yosef, M. Interlandi, et al., Mapping single-cell data to reference atlases by transfer learning, *Nat. Biotechnol.*, vol. 40, no. 1, pp. 121–130, 2022.

[50] A. Gayoso, R. Lopez, G. Xing, P. Boyeau, V. V. P. Amiri, J. Hong, K. Wu, M. Jayasuriya, E. Mehlman, M. Langevin, et al., A python library for probabilistic analysis of single-cell omics data, *Nat. Biotechnol.*, vol. 40, no. 2, pp. 163–166, 2022.

[51] K. Siletti, R. Hodge, A. M. Albiach, K. Lee, S. L. Ding, L. Hu, P. Lönnerberg, T. Bakken, T. Casper, M. Clark, et al., Transcriptomic diversity of cell types across the adult human brain, *Science*, vol. 382, no. 6667, p. eadd7046, 2023.

[52] C. Zhu, M. Yu, H. Huang, I. Juric, A. Abnousi, R. Hu, J. Lucero, M. M. Behrens, M. Hu, and B. Ren, An ultra high-throughput method for single-cell joint analysis of open chromatin and transcriptome, *Nat. Struct. Mol. Biol.*, vol. 26, no. 11, pp. 1063–1070, 2019.

[53] S. Chen, B. B. Lake, and K. Zhang, High-throughput sequencing of the transcriptome and chromatin accessibility in the same cell, *Nat. Biotechnol.*, vol. 37, no. 12, pp. 1452–1457, 2019.

[54] X. Han, Z. Zhou, L. Fei, H. Sun, R. Wang, Y. Chen, H. Chen, J. Wang, H. Tang, W. Ge, et al., Construction of a human cell landscape at single-cell level, *Nature*, vol. 581, no. 7808, pp. 303–309, 2020.

[55] V. Thorsson, D. L. Gibbs, S. D. Brown, D. Wolf, D. S. Bortone, T. H. O. Yang, E. Porta-Pardo, G. F. Gao, C. L. Plaisier, J. A. Eddy, et al., The immune landscape of cancer, *Immunity*, vol. 48, no. 4, pp. 812–830.e14, 2018.

[56] Y. R. Peng, K. Shekhar, W. Yan, D. Herrmann, A. Sappington, G. S. Bryman, T. Van Zyl, M. T. H. Do, A. Regev, and J. R. Sanes, Molecular classification and comparative taxonomics of foveal and peripheral cells in primate retina, *Cell*, vol. 176, no. 5, pp. 1222–1237.e22, 2019.

[57] GTEx Consortium, Erratum: Genetic effects on gene expression across human tissues, *Nature*, vol. 553, no. 7689, pp. 530–530, 2018.

[58] The Tabula Muris Consortium, Single-cell transcriptomics of 20 mouse organs creates a Tabula Muris, *Nature*, vol. 562, no. 7727, pp. 367–372, 2018.

[59] Y. Li, P. Ren, A. Dawson, H. G. Vasquez, W. Ageedi, C. Zhang, W. Luo, R. Chen, Y. Li, S. Kim, et al., Single-cell transcriptome analysis reveals dynamic cell populations and differential gene expression patterns in control and aneurysmal human aortic tissue, *Circulation*, vol. 142, no. 14, pp. 1374–1388, 2020.

[60] T. B. Buus, A. Herrera, E. Ivanova, E. Mimitou, A. Cheng, R. S. Herati, T. Papagiannakopoulos, P. Smibert, N. Odum, and S. B. Koralov, Improving oligo-conjugated antibody signal in multimodal single-cell analysis, *eLife*, vol. 10, p. e61973, 2021.

[61] X. Shao, H. Yang, X. Zhuang, J. Liao, P. Yang, J. Cheng, X. Lu, H. Chen, and X. Fan, scDeepSort: A pre-trained cell-type annotation method for single-cell transcriptomics using deep learning with a weighted graph neural network, *Nucleic Acids Res*., vol. 49, no. 21, p. e122–e122, 2021.

[62] J. Hu, X. Li, G. Hu, Y. Lyu, K. Susztak, and M. Li, Iterative transfer learning with neural network for clustering and cell type classification in single-cell RNA-seq analysis, *Nat. Mach. Intell.*, vol. 2, no. 10, pp. 607–618, 2020.

[63] K. Zhang, J. D. Hocker, M. Miller, X. Hou, J. Chiou, O. B. Poirion, Y. Qiu, Y. E. Li, K. J. Gaulton, A. Wang, et al., A single-cell atlas of chromatin accessibility in the human genome, *Cell*, vol. 184, no. 24, pp. 5985–6001.e19, 2021.

[64] M. J. Muraro, G. Dharmadhikari, D. Grün, N. Groen, T. Dielen, E. Jansen, L. Van Gurp, M. A. Engelse, F. Carlotti, E. J. P. De Koning, et al., A single-cell transcriptome atlas of the human pancreas, *Cell Syst.*, vol. 3, no. 4, pp. 385–394.e3, 2016.

[65] A˚. Segerstolpe, A. Palasantza, P. Eliasson, E. M. Andersson, A. C. Andréasson, X. Sun, S. Picelli, A. Sabirsh, M. Clausen, M. K. Bjursell, et al., Single-cell transcriptome profiling of human pancreatic islets in health and type 2 diabetes, *Cell Metab.*, vol. 24, no. 4, pp. 593–607, 2016.

[66] Y. Xin, J. Kim, H. Okamoto, M. Ni, Y. Wei, C. Adler, A. J. Murphy, G. D. Yancopoulos, C. Lin, and J. Gromada, RNA sequencing of single human islet cells reveals type 2 diabetes genes, *Cell Metab*., vol. 24, no. 4, pp. 608–615, 2016.

[67] J. Chen, H. Xu, W. Tao, Z. Chen, Y. Zhao, and J. D. J. Han, Transformer for one stop interpretable cell type annotation, *Nat. Commun.*, vol. 14, no. 1, p. 223, 2023.

[68] L. Schirmer, D. Velmeshev, S. Holmqvist, M. Kaufmann, S. Werneburg, D. Jung, S. Vistnes, J. H. Stockley, A. Young, M. Steindel, et al., Neuronal vulnerability and multilineage diversity in multiple sclerosis, *Nature*, vol. 573, no. 7772, pp. 75–82, 2019.

[69] S. Cheng, Z. Li, R. Gao, B. Xing, Y. Gao, Y. Yang, S. Qin, L. Zhang, H. Ouyang, P. Du, et al., A pan-cancer single-cell transcriptional atlas of tumor infiltrating myeloid cells, *Cell*, vol. 184, no. 3, pp. 792–809.e23, 2021.

[70] I. Vastrik, P. D'Eustachio, E. Schmidt, G. Joshi-Tope, G. Gopinath, D. Croft, B. De Bono, M. Gillespie, B. Jassal, S. Lewis, et al., Reactome: A knowledge base of biologic pathways and processes, *Genome Biol.*, vol. 8, no. 3, p. R39, 2007.

[71] L. Garcia-Alonso, C. H. Holland, M. M. Ibrahim, D. Turei, and J. Saez-Rodriguez, Benchmark and integration of resources for the estimation of human transcription factor activities, *Genome Res.*, vol. 29, no. 8, pp. 1363–1375, 2019.

[72] H. Han, J. W. Cho, S. Lee, A. Yun, H. Kim, D. Bae, S. Yang, C. Y. Kim, M. Lee, E. Kim, et al., TRRUST v2: An expanded reference database of human and mouse transcriptional regulatory interactions, *Nucleic Acids Res*., vol. 46, no. D1, pp. D380–D386, 2018.

[73] M. D. Luecken, M. Büttner, K. Chaichoompu, A. Danese, M. Interlandi, M. F. Mueller, D. C. Strobl, L. Zappia, M. Dugas, M. Colomé-Tatché, et al., Benchmarking atlas-level data integration in single-cell genomics, *Nat. Methods*, vol. 19, no. 1, pp. 41–50, 2022.

[74] Z. Zou, T. Ohta, F. Miura, and S. Oki, ChIP-Atlas 2021 update: A data-mining suite for exploring epigenomic landscapes by fully integrating ChIP-seq, ATAC-seq and bisulfite-seq data, *Nucleic Acids Res*., vol. 50, no. W1, pp. W175–W182, 2022.

[75] B. Adamson, T. M. Norman, M. Jost, M. Y. Cho, J. K. Nuñez, Y. Chen, J. E. Villalta, L. A. Gilbert, M. A. Horlbeck, M. Y. Hein, et al., A multiplexed single-cell CRISPR screening platform enables systematic dissection of the unfolded protein response, *Cell*, vol. 167, no. 7, pp. 1867–1882.e21, 2016.

[76] A. Dixit, O. Parnas, B. Li, J. Chen, C. P. Fulco, L. Jerby-Arnon, N. D. Marjanovic, D. Dionne, T. Burks, R. Raychowdhury, et al., Perturb-seq: Dissecting molecular circuits with scalable single-cell RNA profiling of pooled genetic screens, *Cell*, vol. 167, no. 7, pp. 1853–1866.e17, 2016.

[77] T. M. Norman, M. A. Horlbeck, J. M. Replogle, A. Y. Ge, A. Xu, M. Jost, L. A. Gilbert, and J. S. Weissman, Exploring genetic interaction manifolds constructed from rich single-cell phenotypes, *Science*, vol. 365, no. 6455, pp. 786–793, 2019.

[78] M. Büttner, Z. Miao, F. A. Wolf, S. A. Teichmann, and F. J. Theis, A test metric for assessing single-cell RNA-seq batch correction, *Nat. Methods*, vol. 16, no. 1, pp. 43–49, 2019.

[79] P. J. Rousseeuw, Silhouettes: A graphical aid to the interpretation and validation of cluster analysis, *J. Comput. Appl. Math.*, vol. 20, pp. 53–65, 1987.

[80] R. Navigli and M. Lapata, An experimental study of graph connectivity for unsupervised word sense disambiguation, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 4, pp. 678–692, 2010.

[81] Y. Song, Z. Miao, A. Brazma, and I. Papatheodorou, Benchmarking strategies for cross-species integration of single-cell RNA sequencing data, *Nature Communications*, vol. 14, no. 1, p. 6495, 2023.

[82] R. Satija, J. A. Farrell, D. Gennert, A. F. Schier, and A. Regev, Spatial reconstruction of single-cell gene expression data, *Nat. Biotechnol.*, vol. 33, no. 5, pp. 495–502, 2015.

[83] I. Korsunsky, N. Millard, J. Fan, K. Slowikowski, F. Zhang, K. Wei, Y. Baglaenko, M. Brenner, P. R. Loh, and S. Raychaudhuri, Fast, sensitive and accurate integration of single-cell data with harmony, *Nat. Methods*, vol. 16, no. 12, pp. 1289–1296, 2019.

[84] R. Lopez, J. Regier, M. B. Cole, M. I. Jordan, and N. Yosef, Deep generative modeling for single-cell transcriptomics, *Nat. Methods*, vol. 15, no. 12, pp. 1053–1058, 2018.

[85] K. Z. Kedzierska, L. Crawford, A. P. Amini, and A. X. Lu, Assessing the limits of zero-shot foundation models in single-cell biology, bioRxiv, doi: 10.1101/2023.10.16.561085.

[86] L. Hubert and P. Arabie, Comparing partitions, *J. Classif.*, vol. 2, no. 1, pp. 193–218, 1985.

[87] J. P. W. Pluim, J. B. A. Maintz, and M. A. Viergever, Mutual-information-based registration of medical images: A survey, *IEEE Trans. Med. Imaging*, vol. 22, no. 8, pp. 986–1004, 2003.

[88] R. Argelaguet, D. Arnol, D. Bredikhin, Y. Deloro, B. Velten, J. C. Marioni, and O. Stegle, MOFA+: A statistical framework for comprehensive integration of multi-modal single-cell data, *Genome Biol*., vol. 21, no. 1, p. 111, 2020.

[89] A. Gayoso, Z. Steier, R. Lopez, J. Regier, K. L. Nazor, A. Streets, and N. Yosef, Joint probabilistic modeling of single-cell multi-omic data with totalVI, *Nat. Methods*, vol. 18, no. 3, pp. 272–282, 2021.

[90] W. Lan, D. Lai, Q. Chen, X. Wu, B. Chen, J. Liu, J. Wang, and Y. P. P. Chen, LDICDL: LncRNA-disease association identification based on collaborative deep learning, *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 19, no. 3, pp. 1715–1723, 2022.

[91] W. Lan, M. Liu, J. Chen, J. Ye, R. Zheng, X. Zhu, and W. Peng, JLONMFSC: Clustering scRNA-seq data based on joint learning of non-negative matrix factorization and subspace clustering, *Methods*, vol. 222, pp. 1–9, 2024.

[92] W. Lan, J. Chen, Q. Chen, J. Liu, J. Wang, and Y. P. P. Chen, Detecting cell type from single cell RNA sequencing based on deep bi-stochastic graph regularized matrix factorization, bioRxiv, doi: 10.1101/2022.05.16.492212.

[93] Z. Wan, Y. Mahajan, B. W. Kang, T. J. Moore, and J. H. Cho, A survey on centrality metrics and their network resilience analysis, *IEEE Access*, vol. 9, pp. 104773–104819, 2021.

[94] D. W. Huang, B. T. Sherman, and R. A. Lempicki, Bioinformatics enrichment tools: Paths toward the comprehensive functional analysis of large gene lists, *Nucleic Acids Res.*, vol. 37, no. 1, pp. 1–13, 2009.

[95] M. Paczkowska, J. Barenboim, N. Sintupisut, N. S. Fox, H. Zhu, D. Abd-Rabbo, M. W. Mee, P. C. Boutros, PCAWG Drivers and Functional Interpretation Working Group, J. Reimand, et al., Integrative pathway enrichment analysis of multivariate omics data, *Nat. Commun.*, vol. 11, no. 1, p. 735, 2020.

[96] W. Lan, C. Li, Q. Chen, N. Yu, Y. Pan, Y. Zheng, and Y. P. P. Chen, LGCDA: Predicting circRNA-disease association based on fusion of local and global features, *IEEE/ACM Trans. Comput. Biol. Bioinf*., doi: 10.1109/TCBB.2024.3387913.

[97] W. Lan, Y. Dong, Q. Chen, R. Zheng, J. Liu, Y. Pan, and Y. P. P. Chen, KGANCDA: Predicting circRNA-disease associations based on knowledge graph attention network, *Briefings Bioinf*., vol. 23, no. 1, p. bbab494, 2022.

[98] Y. Roohani, K. Huang, and J. Leskovec, GEARS:

Predicting transcriptional outcomes of novel multi-gene perturbations, bioRxiv, doi: 10.1101/2022.07.12.499735.

[99] M. Lotfollahi, A. K. Susmelj, C. De Donno, L. Hetzel, Y. Ji, I. L. Ibarra, S. R. Srivatsan, M. Naghipourfar, R. M. Daza, B. Martin, et al., Predicting cellular responses to complex perturbations in high-throughput screens, *Mol. Syst. Biol.*, vol. 19, no. 6, p. e11517, 2023.

[100] M. Lotfollahi, F. A. Wolf, and F. J. Theis, scGen predicts single-cell perturbation responses, *Nat. Methods*, vol. 16, no. 8, pp. 715–721, 2019.

[101] W. Lan, X. Sun, Q. Chen, J. Ye, X. Zhu, and Y. Pan, scIAC: Clustering scATAC-seq data based on student's t-distribution similarity imputation and denoising autoencoder, in *Proc. 2022 IEEE Int. Conf. Bioinformatics and Biomedicine* (*BIBM*), Las Vegas, NV, USA, 2022, pp. 206–211.

[102] M. Marouf, P. Machart, V. Bansal, C. Kilian, D. S. Magruder, C. F. Krebs, and S. Bonn, Realistic in silico generation and augmentation of single-cell RNA-seq data using generative adversarial networks, *Nat. Commun.*, vol. 11, no. 1, p. 166, 2020.

[103] A. Katharopoulos, A. Vyas, N. Pappas, and F. Fleuret, Transformers are RNNs: Fast autoregressive transformers with linear attention, in *Proc. 37th Int. Conf. Machine Learning*, Virtual Event, 2020, p. 478.

[104] W. Lan, H. Liao, Q. Chen, L. Zhu, Y. Pan, and Y. P. P. Chen, DeepKEGG: A multi-omics data integration framework with biological insights for cancer recurrence prediction and biomarker discovery, *Briefings Bioinf.*, vol. 25, no. 3, p. bbae185, 2024.

[105] M. Treppner, H. Binder, and M. Hess, Interpretable generative deep learning: An illustration with single cell gene expression data, *Hum. Genet.*, vol. 141, no. 9, pp. 1481–1498, 2022.

[106] T. Liu, K. Li, Y. Wang, H. Li, and H. Zhao, Evaluating the utilities of large language models in single-cell data analysis, bioRxiv, doi: 10.1101/2023.09.08.555192.

[107] A. R. Alsabbagh, A. M. R. De Infante, D. Gomez-Cabrero, N. A. Kiani, S. A. Khan, and J. N. Tegnér, Foundation models meet imbalanced single-cell data when learning cell type annotations, bioRxiv, doi: 10.1101/2023.10.24.563625.

[108] R. Boiarsky, N. Singh, A. Buendia, G. Getz, and D. Sontag, A deep dive into single-cell RNA sequencing foundation models, bioRxiv, doi: 10.1101/2023.10.19.563100.

[109] X. Yang, K. K. Mann, H. Wu, and J. Ding, scCross: Bridging modalities in single-cell multi-omics-seamless integration, cross-modal synthesis, and in-silico exploration, bioRxiv, doi: 10.1101/2023.11.22.568376.

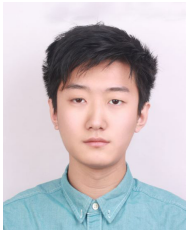
[110] Z. Shi and H. Wu, CTPredictor: A comprehensive and robust framework for predicting cell types by integrating multi-scale features from single-cell Hi-C data, *Comput. Biol. Med.*, vol. 173, p. 108336, 2024.

Wei Lan received the PhD degree in computer science from Central South University, China in 2016. Currently, he is an associate professor at School of Computer, Electronic and Information, Guangxi University, China. His current research interests include bioinformatics and machine learning.

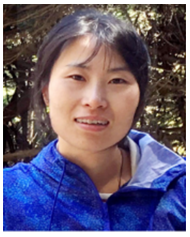
Qingfeng Chen received the PhD degree in computer science from University of Technology Sydney, Australia in 2004. He is currently a professor and the director of bioinformatics team at State Key Laboratory for Conservation and Utilization of Subtropical Agro-Bioresources and School of Computer, Electronic and Information, Guangxi University, China. His research interests include bioinformatics, data mining, and artificial intelligence.

Junyue Cao received the PhD degree in software engineering from University of Chinese Academy of Sciences, China in 2021. He is currently a postdoctoral researcher at College of Life Science and Technology, Guangxi University, China. His research interests are focused on the intersection of artificial intelligence and bioinformatics.

Guohang He received the BEng degree in software engineering from Harbin Engineering University, China in 2023. He is currently a master student in computer science and technology at Guangxi University, China. His research interests include bioinformatics and deep learning.



Mingyang Liu received the BEng degree in computer science and technology from Shandong Normal University, China in 2020. He is currently a master student in computer science and technology at Guangxi University, China. His research interests include bioinformatics and deep learning.



Wei Peng received the PhD degree in computer science from Central South University, China in 2013. Currently, she is a professor at Kunming University of Science and Technology, China. Her research interests include bioinformatics and data mining.