BMC Bioinformatics

RESEARCH

Open Access

scFTAT: a novel cell annotation method integrating FFT and transformer

Binhua Tang[1,2,3*] and Yiyao Chen[1,2]

*Correspondence:
bh.tang@hhu.edu.cn

[1] College of Information Science and Engineering, Hohai University, Jiangsu 213200, China
[2] Key Laboratory of Maritime Intelligent Cyberspace Technology (Hohai University), Ministry of Education, Jiangsu 213200, China
[3] BGI Research, Changzhou 213299, Jiangsu, China

Abstract

Background: Advancements in high-throughput sequencing and deep learning have boosted single-cell RNA studies. However, current methods for annotating single-cell data face challenges due to high data sparsity and tedious manual annotation on large-scale data.

Results: Thus, we proposed a novel annotation model integrating FFT (Fast Fourier Transform) and an enhanced Transformer, named scFTAT. Initially, it reduces data sparsity using LDA (Linear Discriminant Analysis). Subsequently, automatic cell annotation is achieved through a proposed module integrating FFT and an enhanced Transformer. Moreover, the model is fine-tuned to improve training performance by effectively incorporating such techniques as kernel approximation, position encoding enhancement, and attention enhancement modules. Compared to existing popular annotation tools, scFTAT maintains high accuracy and robustness on six typical datasets. Specifically, the model achieves an accuracy of 0.93 on the human kidney data, with an F1 score of 0.84, precision of 0.96, recall rate of 0.80, and Matthews correlation coefficient of 0.89. The highest accuracy of the compared methods is 0.92, with an F1 score of 0.71, precision of 0.75, recall rate of 0.73, and Matthews correlation coefficient of 0.85. The compiled codes and supplements are available at: https://github.com/gladex/scFTAT.

Conclusion: In summary, the proposed scFTAT effectively integrates FFT and enhanced Transformer for automatic feature learning, addressing the challenges of high sparsity and tedious manual annotation in single-cell profiling data. Experiments on six typical scRNA-seq datasets from human and mouse tissues evaluate the model using five metrics as accuracy, F1 score, precision, recall, and Matthews correlation coefficient. Performance comparisons with existing methods further demonstrate the efficiency and robustness of our proposed method.

Keywords: Single cell, Cell type identification, Deep learning, Fast Fourier Transform, Transformer

Background

Single-cell RNA sequencing technology has become widely applicable over decades [1]. Its high throughput and sensitivity allow for comprehensive profiling of individual cell state changes, enabling the analysis of cellular heterogeneity across many cells and

increasing the possibility of discovering hidden cell populations. Although a few annotation methods have been developed so far, cell type identification remains as an essential but unresolved challenge in single-cell studies [2]. Those methods can be mainly categorized as manual annotation with predefined marker genes and automated annotation via machine learning techniques [3].

Manual annotation via markers further covers two approaches. One approach involves manually labeling the single-cell data type through unsupervised clustering [4]. Nevertheless, the scale of single-cell data has experienced exponential growth due to improved cell profiling, rendering manual labeling of large-scale data extremely laborious. Moreover, manual labeling heavily depends on prior knowledge, potentially resulting in unpredictable experimental errors [5], and no specific marker is usually available for referencing an unknown cell type. The other approaches are less dependent on markers compared to the previous one. It primarily establishes classification by analyzing the correlation of gene expression between query and reference samples within single-cell data. Popular analysis tools include Seurat [6], Scanpy [7], and SingleR [8]. However, these methods also tend to be impacted by the batch effect when conducting experiments based on cross-platform datasets [9].

The state-of-the-art methods typically integrate deep learning techniques into annotating single-cell data. Supervised models are generally trained on labeled data and then predict unknown data. Contrarily, unsupervised models learn features through such deep structures as autoencoder-based VAE [10], graph network-based GNN [11], and more recent Transformers [12]. DCA, an AE improvement based on ZINB loss, addressed the dropout issue in preprocessing single-cell data [13]. Based on DCA, variant AE algorithms like scziDesk [14] and scDeepCluster [15] further optimized the model training and performance. And, scVI integrates the batch normalization modules into a standard VAE framework to address batch effects [16]. Furthermore, scDSC, improved on graph neural networks, employs a semi-supervised approach to unify AE and GNN for cell clustering [17].

Despite the progress made so far, a few crucial but challenging problems still exist. Most methods require highly variable genes (HVGs) for cell annotation, and the detected HVGs may vary with different batches and datasets, resulting in discrepancies in model performance [18]. Secondly, solely focusing on HVGs may overlook the gene correlation between HVGs and non-HVGs, potentially leading to the loss of valuable information in non-HVGs that is beneficial for novel cell types [19].

To address the above challenges, we propose an integrated cell annotation approach, scFTAT, leveraging FFT and enhanced Transformer to perform multi-class clustering on scRNA-seq data. The approach first preprocesses scRNA-seq data with LDA rather than commonly identifying HVGs. Meanwhile, instead of feeding into the Transformer directly [20], it introduces an FFT-based module to encode the inputs; subsequently, the encoding segment and attention scoring segment [21] of the self-attention layer in the Transformer are augmented with rotation encoding matrices and kernel approximation, to reduce time complexity. Finally, a parallel structure in the feedforward network segment integrates global and local information to enhance the model representation. Further validation experiments across six typical scRNA-seq datasets and six annotation methods demonstrate our proposed method's exceptional performance and robustness.

Materials and methods

Data preprocessing and summary

This study utilized single-cell RNA-seq data from human and mouse tissues, mainly from the Human Cell Atlas (HCA) [22], comprising 562,977 cells from 56 different tissues. Additionally, the Mouse Cell Atlas (MCA) data involving 201,764 cells from 32 tissues were also selected (https://figshare.com/articles/MCADGE_Data/5435866). Unannotated cells were further excluded during preprocessing. Specifically, we extracted six typical datasets: human bladder, human kidney, human fetal pancreas, mouse bladder, mouse kidney, and mouse spleen.

In preprocessing, cells constituting less than 0.2% of the total cell count were removed, following which 80% of the cell datasets were randomly assigned as the training sets and the remaining 20% as the testing for experiments. Table 1 summarizes the final cell count, gene count, cluster count, and sparsity information for the six collected datasets, respectively. The sparsity ratio refers to the percentage of zero elements in the single-cell expression matrix.

The framework of the proposed scFTAT

As depicted in Fig. 1, the dimensionality reduction layer reduces the training time with LDA while retaining essential features. Then, the data is trained through the FFT encoding and enhanced Transformer layer. The FFT encoder consists of an FFT, a weighted gating, and an inverse FFT (IFFT) layer. The weighted gating layer utilizes trainable weight parameters to determine the frequency weights within the FFT encoding layer.

The improved Transformer layer comprises a multi-head attention (MHA) module and a feedforward network based on two-dimensional attention. The MHA module uses kernel function approximation to replace the original softmax operation. Additionally, rotational position encoding is introduced in the attention score to achieve faster and more efficient relative position encoding and improved residual connection. The enhanced feedforward layer incorporates an attention module that combines global and local information to enhance model generalization. The final classification prediction layer utilizes a linear classifier to predict cell type.

After the original data undergoes LDA, the output passes through three modules: FFT, Transformer, and a classifier. The FFT and IFFT processes are treated as encoding–decoding processes in the FFT encoding module. A trainable matrix W is introduced in the middle layer to enable FFT-based training. Subsequently, a feedforward network and normalization operation are applied to connect to the next module. The

Table 1 Summary of six typical scRNA-seq experiment datasets

Data	Cell #	Gene #	Cluster #	Sparsity ratio
Human-bladder	2750	17,562	12	97.44%
Human-kidney	3849	15,228	15	96.66%
Human-fetal-pancreas	2830	18,210	17	97.57%
Mouse-bladder	2746	19,123	13	94.42%
Mouse-kidney	4682	18,328	17	96.99%
Mouse-spleen	1970	16,332	10	96.65%

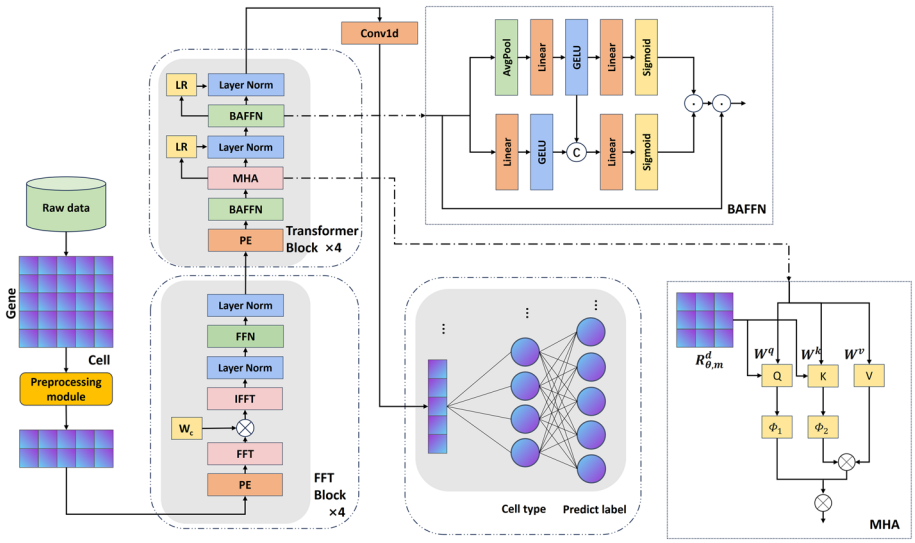

Fig. 1 The workflow framework of the proposed scFTAT, which mainly consists of four components, namely dimensionality reduction, FFT encoding, Transformer encoding, and classification layer, respectively

framework of the enhanced Transformer layer introduces a learnable parameter LR to replace the original residual and normalization in the standard Transformer framework. The MHA module contains a refined unit incorporating relative positional information by multiplying Q and K with a rotation attention matrix after obtaining QKV through the regular process. Additionally, Q and K are approximated using a kernel function to replace the original softmax. The BAFFN module is a specific implementation of the feedforward network that enhances the model generalization by fusing global and local information. The final classification module is implemented directly through a linear classifier based on a one-dimensional CNN.

The dimension reduction layer

Dimensionality reduction is performed using linear discriminant analysis (LDA), which aims to project high-dimensional scRNA-seq expression profiles from a given sample into an optimized lower-dimensional space. To reduce data sparsity, the distances between intra-class samples are minimized, while inter-class distances are maximized.

Thus, exploring the relationship between the original and condensed dimensionality is essential to detect the optimal projection space. Denote the original dimensionality as k, the condensed as k', and the class count as d. Previous studies concluded that the algorithm performs best when k' is set as $d-1$ [23]. This outcome is grounded in the LDA algorithm's theoretical capability to reduce dimensionality to a maximum of $d-1$. Therefore, starting from $d-1$ and gradually increasing the new dimensionality in experiments is preferred.

In practice, to ensure the data after dimensionality reduction are all greater than 0, it is necessary to add a positive definite matrix after the reduction. The specific values of the positive definite matrix need to be adjusted based on the results of dimensionality reduction for different data, depicted as,

$$M' = f_{LDA}(M) + A, A > 0 \tag{1}$$

where M denotes the preprocessed profiling matrix as an input, A is a positive definite matrix with a scalar multiple of the identity matrix, and M' for the resulting matrix after dimension reduction.

The structure of the FFT encoding layer

Following dimensionality reduction, the profiling matrix is fed into the FFT encoding layer, consisting of an FFT network training and a feedforward network layer. Specifically, the FFT network training layer comprises several components, including an FFT layer, a weighted gating layer, and an Inverse FFT layer. The FFT for training facilitates the interaction of information in the frequency domain for the input single-cell data. At the same time, the weighted gating layer employs trainable weight parameters to determine the frequency weights within the FFT encoding layer. In scenarios involving multiple FFT encoding layers, parameters evolve with model changes, and weight parameters can be optimized through backpropagation. The proposed model here employs two-dimensional FFT for data transformation. The previous GFNet directly utilized an FFT-based encoding layer to replace the conventional MHA layer in the Transformer [24]. We place this layer before the attention layer, thereby preserving the advantages of both layers.

In the FFT encoding layer, the feedforward network primarily utilizes the GELU (Gaussian Error Linear Unit) activation function, which helps mitigate the vanishing gradient issue. The feedforward network is denoted as,

$$FFN_{FFT}(x) = GELU(xW_1 + b_1)W_2 + b_2 \tag{2}$$

where x represents the input processed by the FFT network training layer, $xW_1 + b_1$ for the linear transformation layer, and W_2 and b_2 for the weights in the two linear transformation layers.

The structure of the improved Transformer layer

After preliminary training in the frequency domain through the FFT encoding layer, the output is fed into the improved Transformer layer, which generally consists of an MHA module, a feedforward module, and the essential residual layers. Here, the three modules were reconstructed for a more efficient and faster cell annotation.

Given a profiling matrix X as input, an absolute position encoding operation is required before being fed into the MHA layer to preserve its sequential information. Specifically, this involves representing the data position using sine and cosine functions. The details of the Transformer module are given in the supplementary Sect. 1.1. The MHA module is essentially composed of multiple parallel self-attention units, depicted as,

$$Attention(Q, K, V) = SoftMax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{3}$$

where $Q = W^q X_{PE}$, $K = W^k X_{PE}$ and $V = W^v X_{PE}$, W is the linear transformer matrix, Q, K and V for query, key and value vectors, and X_{PE} for the output vector with position encoded, respectively.

Thus, the attention scores are acquired by the inner product of Q and K and then normalized via softmax. The absolute position encoding is directly incorporated into the context representation. However, relative position encoding offers better generalization and scalability than absolute position encoding, which is especially advantageous when dealing with long sequences.

For single-cell data, incorporating the relative information between inputs into the model training can lead to improved performance [25]. Thus, we refine the attention score definition to incorporate relative position encoding, specifically Q and K are further multiplied by a rotation encoding matrix R. And the R and Q metrics in any even dimension are defined as,

$$R^d_{\theta,m} = \begin{pmatrix} \cos m\theta_1 & -\sin m\theta_1 & 0 & 0 & \cdots & 0 & 0 \\ \sin m\theta_1 & \cos m\theta_1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cos m\theta_2 & -\sin m\theta_2 & \cdots & 0 & 0 \\ 0 & 0 & \sin m\theta_2 & \cos m\theta_2 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \cos m\theta_{d/2} & -\sin m\theta_{d/2} \\ 0 & 0 & 0 & 0 & \cdots & \sin m\theta_{d/2} & \cos m\theta_{d/2} \end{pmatrix} \tag{4}$$

$$Q = R^d_{\theta,m}(q^0\, q^1\, \cdots\, q^{d-1})^T \tag{5}$$

where d refers to the spatial dimensionality, m the position of Q, and q the specific Q value at each dimension. And θ can be denoted as, $\Theta = \{\theta_i = 10000^{\frac{-2(i-1)}{d}}, i \in [1, 2, \ldots, \frac{d}{2}]\}$.

Equation (5) represents an orthogonal matrix, and it does not alter the magnitude of the vectors during computation, thereby further ensuring the stability of the model. For $R^d_{\theta,m}$ is sparse, it can be reformulated as,

$$Q = R^d_{\theta,m} X = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_d \end{pmatrix} \otimes \begin{pmatrix} \cos m\theta_1 \\ \cos m\theta_1 \\ \cos m\theta_2 \\ \vdots \\ \cos m\theta_{d/2} \end{pmatrix} + \begin{pmatrix} -x_2 \\ x_1 \\ -x_4 \\ \vdots \\ x_d \end{pmatrix} \otimes \begin{pmatrix} \sin m\theta_1 \\ \sin m\theta_1 \\ \sin m\theta_2 \\ \vdots \\ \sin m\theta_{d/2} \end{pmatrix} \tag{6}$$

The new Q and K derived above contain relative positional information, which means relative position encoding is implemented with absolute position encoding. Furthermore, the softmax operation is specifically approximated with a kernel function to reduce time complexity [26],

$$\Phi_1 \Phi_2 \approx SoftMax\left(\frac{QK^T}{\sqrt{d}}\right) \tag{7}$$

where Φ_1 and Φ_2 denote the updated Q and K. For a general self-attention module with its Q, K, and V of dimension $L \times d$, the computational complexity of Eq. (3) is $O(L^2 \times d)$. While in Eq. (7), the attention matrix is decomposed into the product of a random

nonlinear function with Q and $\underline{K}$, it can enhance encoding efficiency with the complexity reduced to $O(L \times d^2)$, where d is selectable and less than L. Here Φ is defined as,

$$\Phi = \frac{p}{\sqrt{m}} \exp(W^T x - \frac{\|x\|^2}{2}) \tag{8}$$

where p denotes a positive constant, W the product of an input matrix and a random orthogonal matrix, m the dimension of W, and x the input Q or K. The random orthogonal matrix can reduce input dimensionality while retaining inherent features.

Next, we use the multi-head self-attention mechanism to calculate the attention weight of each head and perform parallel operations, detailed in the supplementary Sect. 1.1. The specific number of heads may vary depending on the size and number of data categories. After combining into an MHA module, the general steps are to pass through normalization and residual connection layers. Here, we utilize the Rezero method [27] to rescale the self-attention block. Specifically, the residual connection is represented as follows,

$$X_i\prime = X_i + \alpha_i sublayer(X_i) \tag{9}$$

where X_i and X_i' denote the input and output of an attention module, respectively; α_i refers to a learnable residual weight shared across each MHA module. This parameter is initialized to zero, which causes the gradients of all parameters for the sublayer function in (9) to vanish at the initial training. Then, during the training, it will reach an appropriate value, thereby accelerating the network convergence.

Upon completion of the attention module and residual layer processing, an additional feedforward network is required to obtain the nonlinear features between data further. Here, we introduce a parallel feedforward function module to enhance global and local information via two branches integrated into the standard feedforward function. Like channel attention [28], average pooling is utilized in the global branch to obtain the global representation of the input, followed by a fully connected operation. In contrast, the local information extraction branch directly performs a fully connected operation to extract features. Subsequently, the two branches are passed through a gating unit to obtain the corresponding attention weights, and the output dimensions are aligned with the input dimensions. Overall, this module can enhance its feature representation without increasing computational complexity.

The convolutional layer for classification

Finally, the extracted feature vectors by the improved Transformer layer are further processed by a convolutional network layer and then fed into a linear classifier for cell-type classification. The loss function is defined as,

$$Loss = -\sum_{i=1}^{outputsize} y_i \log \hat{y}_i \tag{10}$$

where y_i denotes the actual value, $\hat{y}_i$ the model prediction.

Selection of performance metrics

Diverse metrics, including accuracy (ACC), precision, recall, F1 score, and Matthews correlation coefficient (MCC), are utilized to evaluate the proposed method's performance systematically.

$$ACC = \frac{TP + TN}{TP + FP + FN + TN} \tag{11}$$

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{12}$$

$$Macro{-}Precision = \frac{1}{k}\sum_{i=1}^{k}\frac{TP}{TP + FP} \tag{13}$$

$$Macro{-}Recall = \frac{1}{k}\sum_{i=1}^{k}\frac{TP}{TP + FN} \tag{14}$$

$$F_1{-}score = \frac{2 \times Macro{-}Precision \times Macro{-}Recall}{Macro{-}Precision + Macro{-}Recall} \tag{15}$$

where *TP*, *FP*, *FN*, and *TN* are True Positives, False Positives, False Negatives, and True Negatives, respectively. *TP* refers to samples that the model predicts as belonging to a specific class, and that indeed have this class as their actual label; *FP* refers to samples that the model predicts as this class but do not have this class as their actual label; *FN* refers to samples that do not have this class as their actual label but are predicted by the model to be this class; *TN* refers to samples that neither the model predicts as this class nor have this class as their actual label.

Results

Computational efficiency and ablation experiments on scFTAT

Standard feature extraction methods were selected for ablation experiments to validate the impact of the LDA approach on the performance of scFTAT. The final experimental results were obtained by averaging the outcomes from multiple tests. The human-bladder dataset is selected as the study case, and Table 2 presents the experimental results.

The comparison methods here include Singular Value Decomposition (SVD), Non-negative Matrix Factorization (NMF), Locally Linear Embedding (LLE), and

Table 2 Ablation experiment of the diverse approaches on the performance of scFTAT

	ACC	F1	Precision	Recall	MCC
SVD	0.90	0.69	0.70	0.69	0.81
NMF	0.73	0.51	0.65	0.57	0.66
LLE	0.68	0.44	0.41	0.42	0.52
Supervised PCA	0.81	0.69	0.73	0.65	0.77
LDA	0.89	0.84	0.87	0.81	0.81

Supervised Principal Component Analysis (PCA). After preprocessing with these methods, the inputs will be fed into the same improved Transformer module. In Table 2, the best result for each metric is highlighted in bold, where the performance of LDA is significantly better than the others, especially for the three metrics, namely F1, Precision, and Recall. The results represent the performance in analyzing small-class data, which indicates that LDA can still yield satisfactory outcomes for cell types with limited categories.

Furthermore, we select the mouse-bladder dataset for comparative analysis to quantify the improved computational efficiency contributed by the modified Transformer into the proposed scFTAT, which incorporates modules such as kernel function approximation.

In Fig. 2A, the network layer counts selected for the experiment are 4, 8, 12, and 16, respectively. The vertical axis values are obtained by dividing the total training time for all epochs by the epoch size, which is uniformly set to 200 in the experiment. scFTAT is the proposed method, while Transformer refers to the other traditional methods. It can be observed that scFTAT demonstrates higher computational efficiency than the conventional Transformers across various layer counts, and this efficiency persists even as the network depth increases.

In Fig. 2B, Recall, F1, and MCC perform optimally at five attention heads, while Precision reaches its highest at eight. However, the Precision fluctuation between the two cases (five and eight attention heads) does not exceed 5%; thus, we select five attention heads as the Transformer structure in the proposed scFTAT.

Furthermore, a systematic ablation study can effectively verify the impact of the FFT module, the improved Transformer module, and a series of fine-tuning modules on the complete model performance. Due to fluctuations in the actual training and testing data, each time a combination of modules is selected in the ablation experiments, multiple tests will be performed to take the average of the measurements as the final results.

In Fig. 3, it can be observed that as additional modules are incorporated, the overall trend of the metrics improves. While the increases in ACC and MCC are modest,

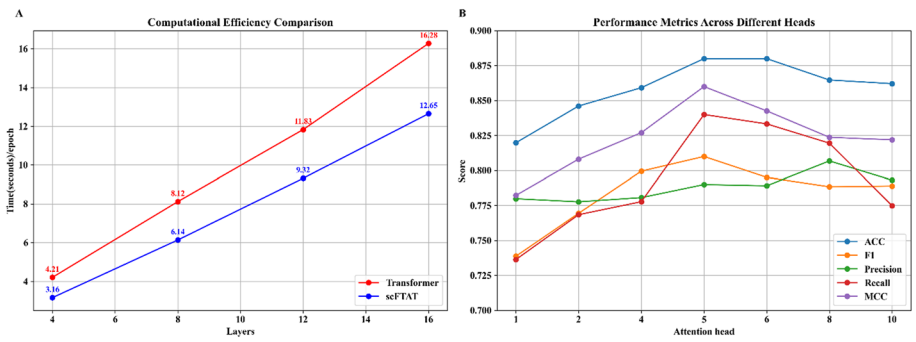

Fig. 2 Comparison of computational efficiency. **A** Epoch time versus network layer between the traditional Transformers and proposed scFTAT. The horizontal axis represents the count of the network layer, while the vertical indicates the running time required in each epoch. **B** Performance versus attention head, the horizontal axis denotes attention head count, and the vertical for ACC, F1, Precision, Recall, and MCC scores, respectively

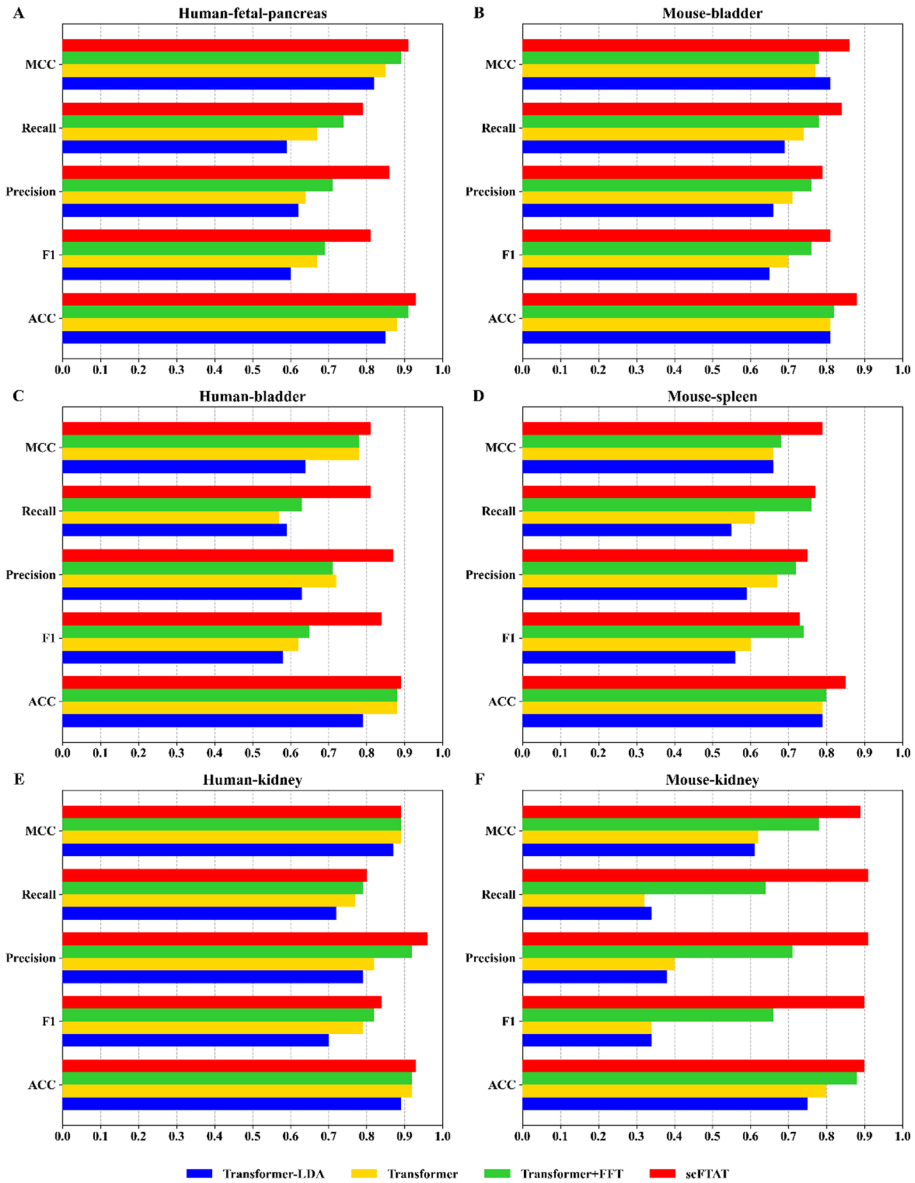

Fig. 3 Ablation experiments of scFTAT across six typical scRNA-seq datasets. **A**, **C** and **E** depict the results on three human tissues, **B**, **D**, and **F** for three mouse tissues. "Transformer-LDA" refers to the configuration where LDA is removed from the Transformer module and replaced with traditional HVGs, "Transformer" for keeping LDA to the previous modules, "Transformer + FFT" for adding an FFT module to the Transformer; "scFTAT" includes the modules above, along with kernel approximation, rotated attention matrix, and other fine-tuning modules to optimize the entire model

F1, Precision, and Recall improve significantly, particularly evident in the mouse-kidney dataset. The comprehensive performance of scFTAT, which integrates all modules, demonstrates the best results on the five metrics across diver human and mouse scRNA-seq data.

Performance evaluation of scFTAT on diverse statistical metrics

To evaluate model performance, the proposed scFTAT is compared with CIForm [20], CellPLM [29], Seurat [6], scDeepSort [22], and PCA-based Transformer. CIForm is based on Transformers, CellPLM is a recent model based on self-supervised learning and Transformer architecture for cell type annotation, and scDeepSort is an annotation approach based on a graph network. The typical Seurat workflow includes PCA- and CCA-based annotation methods; here, the PCA-based was selected in comparison. Additionally, the PCA-Transformer method was included in the comparison to evaluate the performance of the combination of PCA and Transformer.

Figure 4 compares the performance of these six methods across five typical metrics. Both scDeepSort and Seurat-PCA methods exhibit relatively average performance across all experimental datasets. The PCA-Transformer performs better on a specific dataset, while CIForm shows comparable results to scFTAT in some datasets. CellPLM demonstrates reasonable performance in panels (B), (D), and (F), but shows average results in (A), (C), and (E). Generally, our proposed scFTAT delivers the best performance.

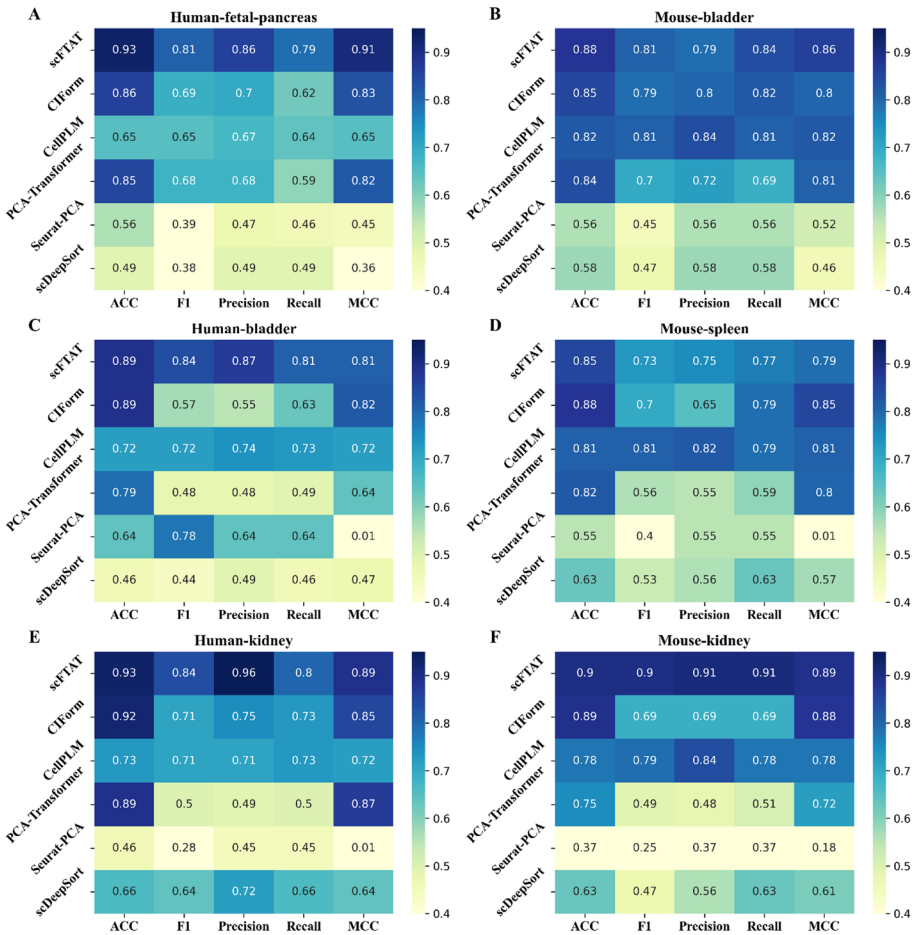

Fig. 4 Performance comparison for the six methods applied to the six typical human and mouse scRNA-seq datasets. **A**, **C** and **E** correspond to the results on the three human tissues, **B**, **D**, and **F** for three mouse tissues, respectively. The evaluation metrics, consistent with the previous ablation experiments, include ACC, F1, Precision, Recall, and MCC

Specifically, regarding ACC, both scDeepSort and Seurat-PCA do not exceed 0.7, while CellPLM ranges between 0.65 and 0.85. The PCA-Transformer maintains above 0.75, while CIForm and the proposed scFTAT consistently achieve above 0.8.

For MCC, scDeepSort and Seurat-PCA exhibit poor performance, with a maximum of 0.6, and CellPLM still fluctuates between 0.65 and 0.85. In contrast, PCA-Transformer, CIForm, and scFTAT perform better, with scFTAT demonstrating the best results overall.

For the F1 score, Precision and Recall, CellPLM exhibits stable performance, with the magnitudes of the three metrics comparable to those of ACC and MCC. scDeepSort and scFTAT perform comparably only in the human bladder data. scFTAT, CIForm, and other comparison methods exhibit a notable performance gap in the different data. Except for the mouse bladder data, where their performances are similar, scFTAT outperforms CIForm in the other data across the three metrics. These metrics indicate that scFTAT performs strongly in cell type identification in the small-class data.

Overall, scFTAT demonstrates remarkable and robust performance on diverse datasets.

Experimental analysis of the proposed scFTAT

We conducted dimensionality reduction and visualization experiments to validate the proposed method's performance by comparing typical methods against scFTAT on single-cell data. The data selected for this experiment is sourced from the mouse kidney dataset, which includes various types of kidney cells, such as distal tubule cells, T cells, macrophages, and others. Figure 5 depicts the detailed visualization results.

The above results demonstrate that the various methods exhibit different performances in cell classification tasks. Both scFTAT and PCA-Transformer (panels A and D) can more clearly differentiate between different types of cell populations, displaying distinct cluster distributions. Moreover, the cluster distribution in scFTAT is notably more compact than that of PCA-Transformer. On the other hand, Seurat-PCA and scDeepSort (panels B and C) show a more dispersed classification. In particular, the scDeepSort method only distinguishes a limited number of cell types with significant overlap. While the Seurat-PCA method identifies a greater variety of cell types, the distribution of similar cell types is more scattered, resulting in a lack of cluster compactness and increasing the complexity of the analysis. These findings align with the earlier comparative experimental results.

Discussion and conclusions

This work proposed a novel single-cell annotation method, scFTAT, based on FFT and a refined Transformer. Structurally, it enhances the Transformer with rotation encoding to incorporate relative positional information better. Meanwhile, it optimizes model training by substituting the softmax layer with kernel function approximation and enhances generalization by improving the Transformer's feedforward module. Ablation and verification experiments demonstrate the effectiveness of the proposed scFTAT for cell-type annotation across various single-cell datasets. Compared to these conventional methods, such as Seurat and scDeepSort, scFTAT exhibits superior annotation performance and robustness.

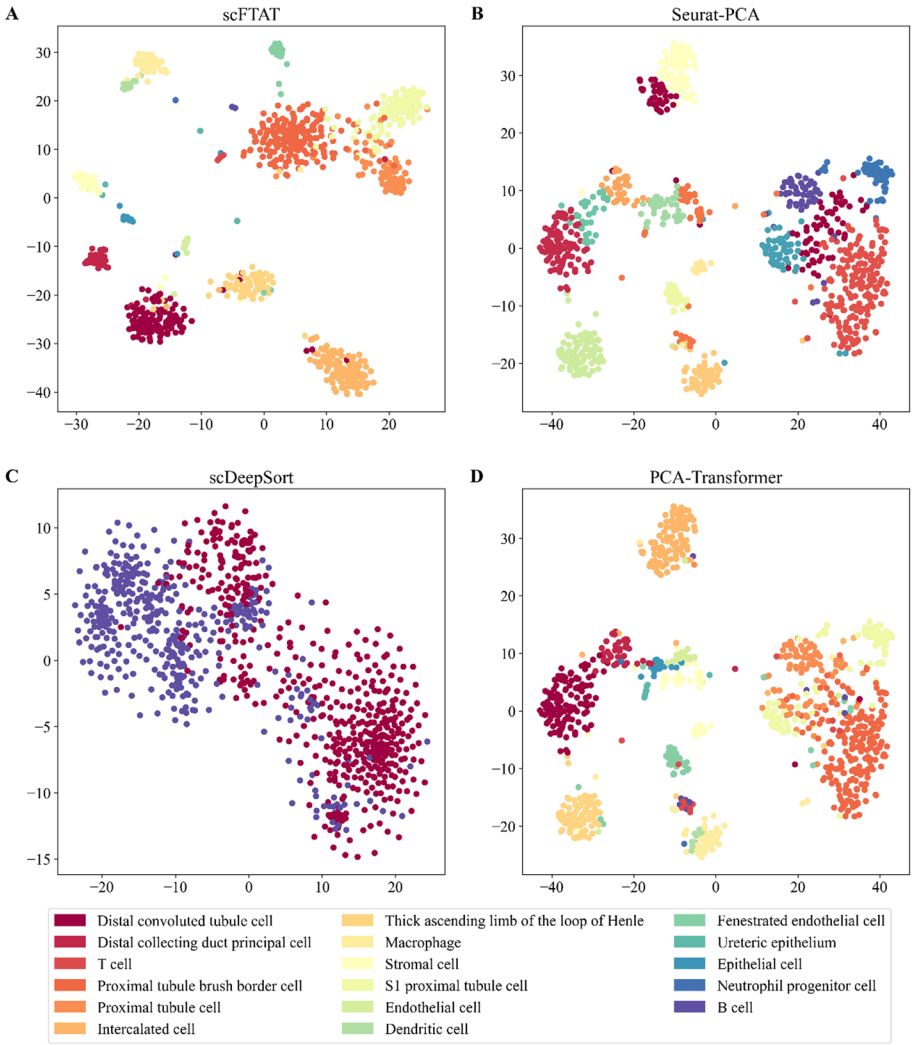

Fig. 5 The visualization results of the tSNE dimensionality reduction for the mouse kidney dataset using different methods. Each point in the figure represents a cell; different colors indicate different cell types. Panels **A**, **B**, **C**, and **D** demonstrate the dimensionality reduction effects using the proposed scFTAT, Seurat-PCA, scDeepSort, and PCA-Transformer, respectively

Although scFTAT has achieved enhanced performance, there are areas for potential improvement. First, the modified Transformer module in scFTAT makes the learning process less interpretable, reflecting the black-box nature of deep learning models. Additionally, its direct data preprocessing and dimensionality reduction methods may result in the loss of inherent features from the original data. Furthermore, reliance on a single data type may impact overall classification performance.

Thus, several promising topics must be further explored to address the abovementioned issues. Firstly, integrating intercellular communication knowledge into a Transformer-based model structure can better reflect the biological process during model training. This enhances feature representation in model generalization, aids in capturing more complex patterns of cellular interactions, and provides a biologically meaningful perspective for downstream analyses.

Next, developing more effective and feasible pretrained models can better capture the key features and gene information from original single-cell data. Recent advancements in NLP and large language models offer promising insights in this area, and a good case in point is automatic annotation using a trained GPT-based model. Additionally, notable progress in deep graph networks, including graph convolutional networks and their variants, enhances single-cell analysis, especially in cell–cell interaction and single-cell spatial omics.

Finally, integrating multi-source single-cell data to enable cross-omics information transfer is anticipated to improve model generalization and overall performance in multidimensional data environments. Multimodal data integration techniques are set to transform the approach to single-cell research, offering new avenues for effectively synthesizing intrinsic data features and their biological significance. Thus, we believe more robust and versatile analytical methods will emerge in multi-source single-cell studies.

Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1186/s12859-025-06061-z.

Additional file 1.

Acknowledgements
The authors sincerely thank the editors and anonymous reviewers for their valuable time and helpful suggestions.

Author contributions
BHT and YYC drafted the manuscript and performed the coding and analysis; BHT led the project and revised the manuscript; both authors proofread and approved the final version.

Funding
The work was partly supported by the research fund (No. B240203012) from the Key Laboratory of Maritime Intelligent Cyberspace Technology (Hohai University), Ministry of Education, China.

Availability of data and materials
The mouse cell atlas dataset involving 201,764 cells from 32 tissues, utilized in the study is available at https://figshare.com/articles/MCADGE_Data/5435866; the self-compiled codes and supplementary materials are available at https://github.com/gladex/scFTAT.

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare no competing interests.

Received: 7 October 2024 Accepted: 22 January 2025
Published online: 25 February 2025

References

1. Shen X, Li X. Deep-learning methods for unveiling large-scale single-cell transcriptomes. Cancer Biol Med. 2024;20(12):972–80.
2. Flores M, et al. Deep learning tackles single-cell analysis-a survey of deep learning for scRNA-seq analysis. Brief Bioinf. 2022;23(1):0531.
3. Pasquini G, et al. Automated methods for cell type annotation on scRNA-seq data. Comput Struct Biotechnol J. 2021;19:961–9.
4. Guo H, Li J. scSorter: assigning cells to known cell types according to marker genes. Genome Biol. 2021;22(1):69.
5. Huang Q, et al. Evaluation of cell type annotation R packages on single-cell RNA-seq data. Genomics Proteom Bioinf. 2021;19(2):267–81.

6. Butler A, et al. Integrating single-cell transcriptomic data across different conditions, technologies, and species. Nat Biotechnol. 2018;36(5):411–20.
7. Wolf FA, Angerer P, Theis FJ. SCANPY: large-scale single-cell gene expression data analysis. Genome Biol. 2018;19(1):15.
8. Aran D, et al. Reference-based analysis of lung single-cell sequencing reveals a transitional profibrotic macrophage. Nat Immunol. 2019;20(2):163–72.
9. Haghverdi L, et al. Batch effects in single-cell RNA-sequencing data are corrected by matching mutual nearest neighbors. Nat Biotechnol. 2018;36(5):421–7.
10. Grønbech CH, et al. scVAE: variational auto-encoders for single-cell gene expression data. Bioinformatics. 2020;36(16):4415–22.
11. Wang J, et al. scGNN is a novel graph neural network framework for single-cell RNA-Seq analyses. Nat Commun. 2021;12(1):1882.
12. Vaswani A, et al. Attention is all you need. In: NIPS. 2017.
13. Eraslan G, et al. Single-cell RNA-seq denoising using a deep count autoencoder. Nat Commun. 2019;10(1):390.
14. Chen L, et al. Deep soft K-means clustering with self-training for single-cell RNA sequence data. Nar Genomics Bioinf. 2020;2(2):0039.
15. Tian T, et al. Clustering single-cell RNA-seq data with a model-based deep learning approach. Nat Mach Intell. 2019;1(4):191–8.
16. Lopez R, et al. Deep generative modeling for single-cell transcriptomics. Nat Methods. 2018;15(12):1053–8.
17. Gan Y, et al. Deep structural clustering for single-cell RNA-seq data jointly through autoencoder and graph neural network. Brief Bioinf. 2022;23(2):bbac018.
18. Alquicira-Hernandez J, et al. scPred: accurate supervised method for cell-type classification from single-cell RNA-seq data. Genome Biol. 2019;20(1):264.
19. Qiu P. Embracing the dropouts in single-cell RNA-seq analysis. Nat Commun. 2020;11(1):1169.
20. Xu J, et al. ClForm as a transformer-based model for cell-type annotation of large-scale single-cell RNA-seq data. Brief Bioinf. 2023;24(4):0195.
21. Liu Y, et al., A survey of visual transformers. IEEE Trans Neural Netw Learn Syst. 2023;1–21.
22. Shao X, et al. scDeepSort: a pre-trained cell-type annotation method for single-cell transcriptomics using deep learning with a weighted graph neural network. Nucleic Acids Res. 2021;49(21): e122.
23. Song T, et al. TransCluster: a cell-type identification method for single-cell RNA-Seq data using deep learning based on transformer. Front Genet. 2022;13:1038919.
24. Rao Y, et al. GFNet: global filter networks for visual recognition. IEEE Trans Pattern Anal Mach Intell. 2023;45(9):10960–73.
25. Liu S, et al. Image classification model based on large kernel attention mechanism and relative position self-attention mechanism. PeerJ Comput Sci. 2023;9: e1344.
26. Katharopoulos A, et al., Transformers are RNNs: fast autoregressive transformers with linear attention. In: D Hal, III, S Aarti (eds) Proceedings of the 37th international conference on machine learning, 2020, PMLR: Proceedings of Machine Learning Research. p. 5156–5165.
27. Bachlechner TC, et al. ReZero is all you need: fast convergence at large depth. In: Conference on uncertainty in artificial intelligence. 2020.
28. Hu J, et al. Squeeze-and-excitation networks. IEEE Trans Pattern Anal Mach Intell. 2020;42(8):2011–23.
29. Liu T, et al., Evaluating the utilities of foundation models in single-cell data analysis. bioRxiv. 2024.

Publisher's Note